Applied Intelligence (2023) 53:18517–18530
https://doi.org/10.1007/s10489-022-04350-1

Multi-label learning with Relief-based label-specific feature selection

Jiadong Zhang[1] · Keyu Liu[1,2] · Xibei Yang[1,3] · Hengrong Ju[4] · Suping Xu[5,6]

Accepted: 16 November 2022 / Published online: 1 February 2023

Abstract
Multi-label learning is an emerging paradigm exploiting samples with rich semantics. As an effective solution to multi-label learning, the strategy of label-specific features (LIFT) has been widely applied. Technically, such strategy feeds the tailored features to learning model instead of the original ones. However, tailoring features for each label may cause redundancy or irrelevance in feature space, thereby deteriorating the learning performance. To alleviate such a problem, a novel multi-label classification method named Relief-LIFT is proposed in this study. Relief-LIFT firstly leverages LIFT to generate the toiled features, and then adjusts Relief to select informative features from those toiled ones for the classification model. Experimental results on 12 real-world multi-label data sets demonstrate that, our proposed Relief-LIFT can achieve better performance as compared with other well-established multi-label classification methods.

Keywords Label-specific feature · Multi-label learning · Feature selection

1 Introduction

The multi-label classification problem is a popular research direction in the field of machine learning [1]. In the existing machine learning paradigm, there are mainly two ways of data labeling: (1) one example is assigned one label; (2) one example is assigned multiple labels. Single-label learning and soft-label learning assume that all examples in the training set are labeled in the first way. Single-label learning aims to answer the basic question, "Which class can describe this example?" Soft-label learning answers another, deeper question, "How does each class describe this example?" i.e., the relative importance of each class to the example. Whether it is single-label learning or soft-label learning, only one label is assigned to each sample while multi-label learning allows training samples to be labeled in the second way. Therefore, multi-label learning can handle ambiguity where an example belongs to multiple labels [2].

In many real-world scenarios, samples are often associated with rich semantics represented by multiple labels. such as society, science, sports and entertainment; in text categorization [3–6], an article can be categorized into multiple topics, such as society, science, sports and entertainment; in image annotation [7–10], an image can be annotated as landscape, ocean, beach and island; in audio classification [11–13], a piece of audio can have several labels, such as vocals, bass, inspirational. Evidently, samples in these applications are always associated with more than one label simultaneously. To learn with multi-label data, a large number of multi-label learning algorithms have been proposed in the past two decades [14–16].

One commonality of these existing methods is that they mainly deal with multi-label learning problems from the perspective of the output space, and the same features inherited from the original input space are directly used to distinguish all labels. Different from these methods, Zhang et al. [17] pioneered the idea of label-specific features (LIFT), and verified its effectiveness and necessity through sufficient experiments. In essence, LIFT abandons the original feature space, while tailoring multiple transformed feature spaces for all the possible labels. It makes sense that in multi-label learning, it is not optimal to use the same feature set for the prediction of each label, and each label should have its unique features. That means, LIFT suggests that specific features have a strong correlation with labels for multi-label learning. However, it does not take into account that constructing class-specific features may encounter an increase in feature dimensionality, and result in a lot of redundant information in the feature space [18]. Consequently, structural information induced by these

✉ Keyu Liu
just_liukeyu@163.com

Extended author information available on the last page of the article.

features may be corrupted, leading to degraded performance of multi-label learning methods [19]. To alleviate this problem, an efficient solution is to perform feature selection on label-specific features. The goal of feature selection is to find the optimal feature subset. By eliminating irrelevant or redundant features, the number of features can be reduced, the accuracy of the model can be improved, and the running time can be reduced [20].

To this end, we propose a novel multi-label learning method based on Relief-based label-specific feature selection (Relief-LIFT). For each possible label, LIFT is firstly leveraged to generate the tailored features which reveal high-level information. Secondly, Relief [20] is modified to select the identified features which capture relevant and necessary information. Finally, multiple binary learners are induced by the tailored yet selected features for multi-label classification. To verify the effectiveness of Relief-LIFT, we conduct comprehensive experiments on 10 real-world multi-label data sets. Experimental studies show that Relief-LIFT has obvious advantages against various multi-label learning algorithms. It is worthwhile to highlight the main contributions of this paper as follows.

- Although the strategy of label-specific features is prevailing and successful for multi-label learning, there is inherent redundancy and irrelevance in the tailored feature spaces. We present an improved Relief feature selection algorithm that remains more informative and identifiable features to simplify the learning model.
- We propose a specific multi-label classification model fed by tailored yet selected features from our improved Relief algorithm, which can effectively improve the learning performance and efficiently reduce the computational cost.
- Experimental studies show that Relief-LIFT has the superior classification performance and time efficiency as compared with other well-established multi-label learning algorithms.

The rest of this paper is organized as following. Section 2 introduces the related work on LIFT for multi-label learning. Section 3 provides the details of Relief-LIFT. Section 4 describes the data sets, evaluation metrics, experimental setup, and then analyzes the results of a comparative study on 12 multi-label data sets. Section 5 concludes and presents possible future research directions.

2 Related work

Before delving into our proposed algorithm Relief-LIFT, we first review related work on multi-label learning.

As reported in [21], existing multi-label learning methods can be divided into two categories: problem transformation methods and algorithm adaptation methods. Problem transformation methods work by transforming a multi-label learning problem into one or more single-label learning problems, which are then solved with a single-label learning algorithm. It follows that, many traditional single-label algorithms can be applied in such algorithms [22–24]. For example, binary relevance (BR) [25] learns a binary classifier for each label independently and predicts each label separately, so it cuts off the relationship between different labels. Label power set (LP) [21] treats each unique label set present in the multi-label training set as a new single-label multi-value class. Although this method takes into account the correlation between different labels, it can easily lead to higher time consumption since the number of new categories grows exponentially with the addition of labels. Algorithm adaptation methods directly deal with multi-label learning by adapting single-label algorithms to the multi-label situation. The process of training a classifier and predicting unseen instances in such algorithms is similar to traditional single-label algorithms. The main advantage of algorithm adaptation methods is that they can exploit the properties of multi-label learning problems in a more compact way. For example, ML-KNN [26] uses the label set of the nearest k samples of the target sample and then determines the label set of the new instance according to the maximum a posteriori probability criterion. MLNB [27] uses feature extraction technology based on principal component analysis to remove irrelevant and redundant features, and then uses feature subset selection technology based on genetic algorithm to select the most appropriate feature subset for prediction.

From the above discussions, it is not difficult to point out that those multi-label algorithms may have their inherent limitations. Traditional problem transformation methods and algorithm adaptation methods use the same features to achieve the purpose of learning different labels. Since the label-specific feature strategy was proposed, more and more scholars have applied this strategy for multi-label learning in recent years. To date, various modifications or extensions of LIFT have been developed due to its successful applications in multi-label learning [28, 29]. To solve the problem that LIFT ignores the positive and negative instance discrimination information, Zhang et al. [30] presented ML-DFL in which a new spectral clustering algorithm was used to extract the close latent structure between positive and negative instances for each class label. Huang et al. [31] directly obtained the label-specific features through feature selection in the original feature space, and proposed LLSF algorithm. Furthermore, in order to solve the problem that the relationship between the original feature space and the label is not close enough, Huang et al. [18] extended the LLSF algorithm and proposed LLSF-DF, which is a multi-label learning algorithm

for label-specific features and class-independent labels. In order to comprehensively consider the compromise between multiple subset evaluation methods in feature selection and optimize the performance of subsets, Zhang et al. [32] proposed a feature selection based on subset evaluation and multi-objective optimization. To address the problem that LIFT does not consider label correlations, Zhang et al. [33] proposed MLFC that adopted the method of jointly learning label-specific features and label correlations, and designed an optimization framework to learn the weight assignment scheme of label-specific features, while considering the correlation between labels by constructing additional features. Considering that LIFT uses the k-means clustering algorithm to learn the exclusive features of markers and can only handle numerical multi-label data, Li et al. [34] proposed a new exclusive feature learning algorithm R-LIFT based on rough set theory.

As reviewed above, existing efforts to enhance LIFT remain in the generation of tailored features. However, they actually ignore the underlying redundancy or irrelevance resulted by these generated features. Needless to say, the learning model fed by these features may be deteriorated. In the next section, an effective approach named Relief-LIFT is proposed which improves LIFT by focusing on these tailored yet informative features via a modified Relief feature selection.

3 Multi-label learning with label-specific feature selection

This section is dedicated to introducing our Relief-LIFT. The notations and technical details are elaborated in the following.

3.1 Basic notations

Let $X = R^{n\times d}$ be the original input space and $L = \{l_1, l_2, ..., l_m\}$ be the finite set of m labels, $F = \{a_1, a_2, ..., a_d\}$ be d features. $T = \{(x_i, Y_i) \mid i = 1, 2, ..., n\}$ denotes the training set with n multi-label training samples, where $x_i \in X$ is a d-dimensional feature vector such that $x_i = [a_1(x_i), a_2(x_i), ..., a_d(x_i)]$, $a_j(x_i)$ represents the eigenvalue of x_i in the a_j feature, Y_i is a set of relevant labels associated with x_i such that $Y_i \subseteq L$. The goal of multi-label learning is to generate a mapping function f: $X \times L \to R$, which is able to correctly predict the label vector of unseen instances.

3.2 Construct label-specific features

LIFT aims to generate tailored features which capture the specific characteristics of each label to facilitate its discrimination process. To achieve this, LIFT first divides the sample into positive and negative domains based on whether the label is present or not, and then uses clustering to form new features. Specifically, with respect to each label l_k, the training samples are divided into two categories, i.e., the set of positive training samples P_k and the set of negative training samples N_k, such that:

$$P_k = \{x_i \mid (x_i, Y_i) \in T, l_k \in Y_i\}, \tag{1}$$

$$N_k = \{x_i \mid (x_i, Y_i) \in T, l_k \notin Y_i\}. \tag{2}$$

In other words, for sample x_i, if the label l_k is its relevant label, it belongs to P_k; otherwise, it belongs to N_k.

In order to capture the specific characteristics of each label, LIFT employs clustering analysis on P_k and N_k, respectively. The k-means algorithm [35] is adopted to partition P_k into m_k^+ disjoint clusters whose clustering centers are denoted by $\{p_1^k, p_2^k, ..., p_{m_k^+}^k\}$. In the same way, N_k is also partitioned into m_k^- disjoint clusters whose centers are denoted as $\{n_1^k, n_2^k, ..., n_{m_k^-}^k\}$. LIFT ensures that the clustering information obtained from positive and negative samples is treated with equal importance, thus sets an equivalent number of clusters for P_k and N_k, i.e., $m_k^+ = m_k^- = m_k$. Specifically, the number of clusters for both positive samples and negative samples are:

$$m_k = \lceil \delta \cdot \min(|\ P_k\ |, |\ N_k\ |) \rceil, \tag{3}$$

where $|\cdot|$ represents the cardinality of a set, $\delta \in [0, 1]$ is the ratio parameter for controlling the number of clusters.

The above two groups of clustering centers describe the internal structures of positive sample P_k and negative sample N_k. The distances between all samples and the two groups of clustering centers can be used to construct label-specific features in the following form:

$$x_i^k = \left[d\left(x_i, p_1^k\right), ..., d\left(x_i, p_{m_k}^k\right), d\left(x_i, n_1^k\right), ..., d(x_i, n_{m_k}^k) \right], \tag{4}$$

where $d(\cdot, \cdot)$ represents the Euclidean distance between two samples. Intuitively, LIFT creates a new label-specific feature space with $2m_k$ features denoted as $LIFT_k = \{a_1^k, a_2^k, ..., a_{m_k}^k, a_{m_k+1}^k, ..., a_{2m_k}^k\}$. In addition, for the label l_k, a label-specific binary training set can be obtained such that:

$$T_k = \left\{ \left(x_i^k, Y_i^k\right) \mid (x_i, Y_i) \in T \right\}, \tag{5}$$

where $Y_i^k = +1$ if $l_k \in Y_i$; otherwise, $Y_i^k = -1$.

3.3 Select label-specific features

The construction of label-specific features will encounter the problem of increasing feature dimensions and generating redundant information. Feature selection can eliminate

irrelevant or redundant features, so as to reduce the number of features, improve the accuracy of the model and reduce the running time, and select the truly relevant feature simplification model.

An improved Relief is proposed to select those tailored features by LIFT with relevance such that $Relief\text{-}LIFT_k \subseteq LIFT_k = \left\{a_1^k, a_2^k, ..., a_{m_k}^k, a_{m_k+1}^k, ..., a_{2m_k}^k\right\}$. As a typical and efficient feature selection algorithm, Relief evaluates the quality of candidate features by the near-hit and near-miss sample. Conceptually, the near-hit sample is the nearest within-class sample, while the near-miss sample is the nearest between-class sample. In this paper, for sample x_i^k, its near-hit and near-miss sample related to feature a_j^k can be formulated as:

$$NH_{ij}^k = \arg\min_{x_m} \left\{ d_j\left(x_i^k, x_m^k\right), Y_i^k = Y_m^k \right\}, \tag{6}$$

$$NM_{ij}^k = \arg\min_{x_m} \left\{ d_j\left(x_i^k, x_m^k\right), Y_i^k \neq Y_m^k \right\}, \tag{7}$$

where $d_j\left(x_i^k, x_m^k\right)$ indicates the Euclidean distance between on the generated feature a_j^k.

Such pair of near-hit and near-miss samples enables us to evaluate all the features in $LIFT_k = \left\{a_1^k, a_2^k, ..., a_{m_k}^k, a_{m_k+1}^k, ..., a_{2m_k}^k\right\}$. Specifically, for sample x_i^k, we assign weight W_{ij}^k to feature a_j^k followed by the formulation:

$$W_{ij}^k = diff_j\left(x_i^k, NM_{ij}^k\right)^2 - diff_j\left(x_i^k, NH_{ij}^k\right)^2, \tag{8}$$

where $diff_j\left(x_i^k, x_m^k\right) = a_j^k\left(x_i^k\right) - a_j^k\left(x_m^k\right)$ returns the difference between two samples on feature a_j^k. The value of W_j^k is referred to as the correlation level of feature a_j^k can be formulated as:

$$W_j^k = \sum_{i=1}^{n} W_{ij}^k. \tag{9}$$

Here, a correlation threshold τ is required to further examine the correlation levels of these features. Specifically, if the correlation level of a feature is greater than τ, then the feature is statistically relevant to the class, and we will select it into the final output feature subset; otherwise, the feature is statistically irrelevant to the class, and we will delete it. Eventually, we can derive a label-specific feature subset, i.e., $Relief\text{-}LIFT_k$.

3.4 Induce classification models

In principle, the classifiers induction process of Relief-LIFT is similar to LIFT: decomposing the multi-label learning problem by breaking it down into several independent binary classification problems. The main difference is that Relief-LIFT induces the classifiers through identifying the label-specific features rather than directly using then without any examination. Specifically, Relief-LIFT induces a family of m classification models $\{f_1, f_2, ..., f_m\}$ in the constructed label-specific feature spaces $Relief\text{-}LIFT_k (1 \leq k \leq m)$. Any binary learner can be employed to induce a classification model f_k for l_k through binary training set. In form, for each $l_k \in L$, let $Relief\text{-}LIFT_k = \{a_1^{*k}, a_2^{*k}, ..., a_s^{*k}\}$ be the selected feature subset from the label-specific features where $s \leq 2m_k$, we can project any sample into a lower dimensional space by $Relief\text{-}LIFT_k$ such that:

$$x_i^{*k} = \left[a_1^{*k}(x_i), a_2^{*k}(x_i), ..., a_s^{*k}(x_i)\right]. \tag{10}$$

Correspondingly, we can obtain a reduced binary training set for l_k represented by:

$$T_k^* = \left\{\left(x_i^{*k}, Y_i^k\right) \mid \left(x_i^k, Y_i\right) \in T_k\right\}. \tag{11}$$

For an unseen sample $x' \in X$, the prediction label set is:

$$Y' = \left\{l_k \mid f\left(\left(x_i^{*k}\right), l_k\right) > 0, 1 \leq k \leq m\right\}. \tag{12}$$

Formally, Relief-LIFT can be designed in the following.

The time complexity of Relief-LIFT mainly comprises of three components: clustering on P_k and N_k in Step 3, the cost of performing clustering on P_k and N_k using k-means is $O(m_k(t_1 \mid P_k \mid + t_2 \mid N_k \mid))$, where t_1 and t_2 are the iterations of k-means on P_k and N_k, respectively. Forming the label-specific feature space in Step 4, forming the label-specific feature space requires $O(2m_k \mid T \mid)$. And feature selection for the label-specific feature space in Steps 6 to 16, the number of features in a specific feature space is $2m_k$, and the improved Relief algorithm calculates the score of n samples for each feature, finally, the time complexity of feature selection is $O(2m_k \times n)$. Therefore, in general the time complexity of Relief-LIFT is $O(m_k(t_1 \mid P_k \mid + t_2 \mid N_k \mid) + 2m_k \mid T \mid + 2m_k \times n)$.

3.5 Example

In the following, we consider showing the detailed algorithm flow on a simple multi-label data set. For a better interpretation, a synthetic multi-label data containing 5 samples is exhibited in Table 1. It can be seen that each sample is described by 3 features, i.e., $F = \{a_1, a_2, a_3\}$, and associated with 3 labels, i.e., $L = \{l_1, l_2, l_3\}$. Furthermore, $Y_1 = \{l_2, l_3\}$ is a set of relevant labels associated with x_1 indicating that l_1 is an irrelevant label to x_1, while l_2 and l_3 are the relevant ones.

We first normalize the multi-label data set in the preprocessing step. Herein, we mainly present the label-specific features of l_1. Since l_1 is relevant label to samples x_2, x_3, x_4, x_5, the positive samples $P_1 = \{x_2, x_3, x_4, x_5\}$ and $N_1 = \{x_1\}$. Assume that $\delta = 0.2$, the number of clusters

Input: The multi-label training set T, the ratio parameter δ for controlling the number of clusters, the threshold τ for feature selection, the unseen sample x'.
Output: The predicted label set Y'.
for k from 1 to m **do**
// Phase I: Construct label-specific features
Form the set of positive samples P_k and the set of negative samples N_k based on T according to (1) and (2);
Perform k-means clustering on P_k and N_k, each with m_k clusters as defined in (3);
Form a label-specific feature space for label l_k with $2m_k$ features denoted as $LIFT_k$ according to (4);
// Phase II: Select label-specific features
Initialize $Relief\text{-}LIFT_k = LIFT_k$;
for j from 1 to $2m_k$ **do**
for i from 1 to n **do**
Find the feature a_j^k related near-hit NH_{ij}^k and near-miss NM_{ij}^k for sample x_i^k according to (6) and (7);
Calculate W_{ij}^k according to (8);
end for
Calculate W_j^k according to (9);
if $W_j^k \leq \tau$ **then**
$Relief\text{-}LIFT_k = Relief\text{-}LIFT_k - a_j^k$;
end if
end for
end for
// Phase III: Induce classification models
for k from 1 to m **do**
Construct the binary training set T_k^* according to (11);
Induce the classification model f_k by invoking any binary learner on T_k^*;
end for
return The predicted label set Y' according to (12).

Algorithm 1 Relief-LIFT.

$m_1 = 1$ according to (3). Correspondingly, we perform 1-means clustering on P_1 and N_1 respectively. It follows that two clustering centers $p_1^1 = [0.3979, 0.4693, 0.2448]$ and $n_1^1 = [0.0816, 0.5306, 0.3469]$ can be obtained. Through calculating the distances of five samples to p_1^1, n_1^1, the label-specific features can be obtained. For instance, $a_1^1(x_1^1) = 0.0380$ and $a_1^2(x_1^1)$ is obtained by the distance of $x1$ to p_1^1 and n_1^1, i.e., $d(x_1, p_1^1) = 0.0380$ and $d(x_1, n_1^1) = 0$. Similarly, the final tailored features of l1 can be derived as shown in Table 2. The near-hit of sample x_1^1 in a_1^1 is itself (Since the negative sample of l_1 is only x_1^1, the near-hit of x_1^1 is itself), and the near-miss is x_4^1. According to the (8), the score W_{11}^1 of sample x_1^1 in a_1^1 is 0.0108. The score of each sample is calculated in turn, and the score W_1^1 of feature a_1^1 is 0.0678 according to the (9). After W_1^1 and W_2^1 are calculated, the features with a score greater than the threshold τ are selected to form the final feature set about l_1.

Table 1 A multi-label data set

Samples	a_1	a_2	a_3	Y
x_1	1.2	3.4	2.5	l_2, l_3
x_2	2.3	2.8	1.7	l_1, l_2
x_3	5.3	5.7	1.6	l_1, l_3
x_4	1.5	1.8	0.8	l_1
x_5	1.9	2.1	3.9	l_1, l_2, l_3

4 Experimental analysis

4.1 Data sets

To demonstrate the effectiveness of our proposed multi-label learning method, 10 real-world multi-label data sets are used in this paper. For each data set $S = \{(x_i, Y_i) \mid i = 1, 2, ..., p\}$, we use symbol $\mid S \mid$, $dim(S)$, $L(S)$ and $F(S)$ to represent the number of samples, number of features, number of relevant labels and feature type respectively. In addition, in order to better describe the characteristics of the data sets, several other multi-label properties are adopted [19], including:

- $LCard(S) = \frac{1}{p}\sum_{i=1}^{p} \mid Y_i \mid$: measures the average number of labels per sample;
- $LDen(S) = \frac{LCard(S)}{L(S)}$: normalizes $LCard(S)$ with the number of possible labels;
- $DL(S) = \mid \{Y_i \mid (x_i, Y_i) \in S\} \mid$: counts the number of different label combinations in S;
- $PDL(S) = \frac{DL(S)}{\mid S \mid}$: normalizes $DL(S)$ with the number of samples.

Table 3 summarizes some detailed characteristics of the multi-label data sets used in the experiment and sorts them in descending order of $\mid S \mid$. The 12 multi-label data sets cover five distinct practical application domains, including music, biology, text, image and chemistry.

Table 2 Tailored features of l_1

Samples	a_1^1	a_2^1
x_1^1	0.3380	0
x_2^1	0.1262	0.3034
x_3^1	0.7477	0.9768
x_4^1	0.4421	0.4804
x_5^1	0.4713	0.4152

Table 3 Characteristics of the experimental data sets

Data sets	$\mid S \mid$	$dim(S)$	$L(S)$	$F(S)$	$LCard(S)$	$LDen(S)$	$DL(S)$	$PDL(S)$	Domain
flag	194	19	7	nominal	3.392	0.485	54	0.096	text
CAL500	502	68	174	numeric	26.044	0.202	502	1.000	music
emotions	593	72	6	numeric	1.869	0.311	27	0.046	music
genbase	662	1185	27	nominal	1.252	0.046	32	0.048	biology
medical	978	1449	45	nominal	1.245	0.028	94	0.096	text
water_quality	1060	16	14	numeric	5.073	0.362	165	0.473	chemistry
enron	1702	1001	53	nominal	3.378	0.064	753	0.442	text
image	2000	294	5	numeric	1.236	0.247	20	0.010	image
scene	2407	294	6	numeric	1.074	0.179	15	0.006	image
yeast	2417	103	14	numeric	4.237	0.303	198	0.082	biology
corel5k	5000	499	374	nominal	3.522	0.009	3175	0.635	image
bibtex	7395	1836	159	nominal	2.402	0.015	2856	0.386	text

4.2 Configuration

Since each sample in a multi-label data set is associated with multiple labels simultaneously, performance evaluation in multi-label learning is more complex than traditional single-label learning [36]. Some popular evaluation metrics in single-label learning system, such as accuracy, precision, recall rate and F-measure, cannot be well adapted to multi-label learning system. Consequently, five commonly used in multi-label learning are adopted in this paper, i.e., average precision [37], coverage [38], hamming loss [39], one error [40] and ranking loss [26].

Given a multi-label test set $T' = \{(x_i, Y_i) \mid 1 \le i \le t\}$, the real-valued function $f(\cdot,\cdot)$ generated by the multi-label learning system can be converted into a ranking function $rank_f(\cdot,\cdot)$. For each $l \in L$, $rank_f(x_i, l)$ maps $f(x_i, l)$ to the grades $\{1, 2, ..., m\}$, i.e., for $f(x_i, l) > f(x_i, l')$, $rank_f(x_i, l) < rank_f(x_i, l')$ holds [26]. The detailed multi-label evaluation metrics are shown below.

- Average precision: evaluates the average fraction of labels ranked above a particular label $l \in Y_i$ which is actually in Y_i. The larger the value of $AveragePrecision(f)$, the better the performance. When $AveragePrecision(f) = 1$ performance is perfect.

$$AveragePrecision(f) = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{\mid Y_i \mid} \times \sum_{l\in Y_i}\frac{\mid \{l' \mid rank_f(x_i, l') \le rank_f(x_i, l), l' \in Y_i\} \mid}{rank_f(x_i, l)}. \quad (13)$$

- Coverage: evaluates how far we need to go down the label list on average to cover all the correct labels for the sample. The smaller the value of $Coverage(f)$, the better the performance. When $Coverage(f) = 0$ performance is perfect.

$$Coverage(f) = \frac{1}{t}\sum_{i=1}^{t}\max_{l\in Y_i} rank_f(x_i, l) - 1. \quad (14)$$

- Hamming loss: evaluates the number of times the sample label pair was misclassified, i.e., predicts an irrelevant label or misses a relevant label for samples. The smaller the value of $HammingLoss(h)$, the better the performance. When $HammingLoss(h) = 0$ performance is perfect.

$$HammingLoss(h) = \frac{1}{t}\sum_{i=1}^{t}\mid h(x_i) \bigotimes Y_i \mid, \quad (15)$$

where $h(x_i)$ is the prediction label set associated with x_i, and $\bigotimes$ represents the symmetric difference between the two sets.

- One error: evaluates how many times a top-ranked label is not in the proper label set for the sample. The smaller the value of $OneError(f)$, the better the performance. When $OneError(f) = 0$ performance is perfect.

$$OneError(f) = \frac{1}{t}\sum_{i=1}^{t}\Psi\left(\left[\arg\max_{l\in Y_i} f(x_i, l)\right] \notin Y_i\right), \quad (16)$$

where for any predicate ξ, $\Psi(\xi) = 1$ if ξ holds; otherwise, $\Psi(\xi) = 0$.

- Ranking loss: evaluates the average fraction for label pairs that are reversely ordered in the sample. The smaller the value of $RankingLoss(f)$, the better the performance. When $RankingLoss(f) = 0$ performance is perfect.

$$RankingLoss(f) = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{\mid Y_i \mid \mid \overline{Y_i} \mid} \mid \{(l, l') \mid f(x_i, l) \le f(x_i, l'), (l, l') \in Y_i \times \overline{Y_i}\} \mid, \quad (17)$$

where $\overline{Y_i}$ denotes the complementary set of Y_i.

In this paper, our Relief-LIFT is compared with four well-established multi-label learning algorithms, the detailed multi-label learning algorithms are shown below.

- MLNB [27]: the method firstly applies a feature extraction technique based on principal component analysis to remove irrelevant and redundant features. After that, a feature subset selection technique based on genetic algorithm is used to select the most suitable feature subset for prediction.
- FRS-SS-LIFT [19]: the method uses fuzzy rough sets to achieve label-specific feature reduction, and combines sample selection to reduce time consumption. We adjust the parameter δ by increasing it from 0.1 to 1.0 (stepsize 0.1), and finally assign δ to 0.2.
- k-MAR [41]: the method uses k pairs of boundary samples to calculate the evaluation function, and establishes the function, the definition of reduction and the design of the algorithm by maximizing the evaluation. we select the appropriate k for each different multi-label data according to the parameter configuration recommended in the literature.
- BILAS [42]: the method generates a customized set of features for a pair of class labels through heuristic prototype selection and embedding. Then, the class predictions induced by the BiLabel-specific features are ensemble to determine the relevance of each class label to unseen instances. The number of clusters parameter τ and the balancing parameter s between separability and dispersion are set to be 0.5 and 0.1.
- MLFE [43]: this paper first characterizes the underlying structure of the feature space by sparse reconstruction between training samples. Second, the reconstruction information is passed from the feature space to the label space, thereby enriching the original categorical labels into numerical labels. The parameters β_1, β_2 and β_3 are selected as 2, 10 and 1 for experiments in this paper, respectively.

4.3 Results

In our experiments, we use 10-fold cross-validation (10-CV) to evaluate the effectiveness of different methods. 10-CV divides all samples into 10 groups of the same size, nine groups constitute the multi-label training set, and one group constitutes the multi-label test set. The classification process is repeated 10 times in turn, and the mean and standard deviation of the 10 experimental results are recorded. Tables 4-14 respectively shows the performance comparison of our method under five parameters and the performance comparison with other multi-label learning methods on the above 12 data sets. The best performers in each category are highlighted in bold. All the experiments are carried out on a personal computer with Windows 10, Intel Core i7-11800H CPU(2.30 GHz) and 16.00 GB memory.

4.3.1 Parametric experiment

Setting the threshold τ of feature selection in Relief-LIFT has a great influence on the result. As shown in Tables 4, 5, 6, 7 and 8, we tune the parameter by increasing the parameter τ from mean(W)-0.1 to mean(W)+0.1 (step size 0.05), where mean(W) is the mean of the weights W of all features. Then a suitable parameter τ for each multi-label data set is employed in the experiment. For example, in the bibtex multi-label data set, Relief-LIFT achieves the optimal performance with respect to three evaluation

Table 4 Experimental results of threshold variation (average ± standard deviation) in terms of average precision (larger values indicate better performance)

Data sets	$mean(W) - 0.1$	$mean(W) - 0.05$	$mean(W)$	$mean(W) + 0.05$	$mean(W) + 0.1$
flag	0.8007±0.0476	0.7985±0.0442	**0.8030±0.0353**	0.7877±0.0491	0.7986±0.0290
CAL500	0.4966±0.0104	0.4969±0.0105	0.4969±0.0118	0.4982±0.0053	**0.4983±0.0161**
emotions	0.5692±0.0151	0.5723±0.0380	**0.5728±0.0262**	0.5675±0.0310	0.5702±0.0283
genbase	0.9973±0.0037	**0.9978±0.0034**	0.9890±0.0060	0.4439±0.0346	0.4426±0.0462
medical	0.9138±0.0182	0.9328±0.0163	**0.9522±0.0183**	0.8437±0.0174	0.8107±0.0165
water_quality	0.5271±0.0361	**0.5512±0.0461**	0.4971±0.0471	0.4821±0.0461	0.3815±0.0436
enron	0.8573±0.0199	0.8522±0.0214	**0.8591±0.0237**	0.8434±0.0150	0.8421±0.0219
image	0.8317±0.0226	**0.8384±0.0174**	0.8149±0.0176	0.5189±0.0152	0.5126±0.0291
scene	**0.8912±0.0141**	0.8900±0.0226	0.8756±0.0127	0.4375±0.0166	0.4359±0.0139
yeast	**0.7772±0.0158**	0.7768±0.0176	0.7682±0.0224	0.7029±0.0188	0.7029±0.0159
corel5k	**0.3802±0.0189**	0.3731±0.0240	0.2671±0.0447	0.1937±0.0237	0.2072±0.0075
bibtex	0.6102±0.0068	**0.6108±0.0092**	0.5713±0.0122	0.1389±0.0076	0.1375±0.0084

Table 5 Experimental results of threshold variation (average ± standard deviation) in terms of coverage (smaller values indicate better performance)

Data sets	$mean(W) - 0.1$	$mean(W) - 0.05$	$mean(W)$	$mean(W) + 0.05$	$mean(W) + 0.1$
flag	0.5505±0.0567	0.5547±0.0392	**0.5449±0.0637**	0.5619±0.0520	0.5625±0.0438
CAL500	0.7466±0.0204	0.7506±0.0209	0.7476±0.0173	0.7441±0.0255	**0.7434±0.0312**
emotions	0.5258±0.0310	**0.5197±0.0291**	0.5226±0.0267	0.5315±0.0401	0.5249±0.0359
genbase	**0.0118±0.0060**	0.0122±0.0059	0.0160±0.0047	0.1812±0.0214	0.1823±0.0216
medical	**0.1273±0.0694**	0.1295±0.0638	0.1294±0.0619	0.1375±0.0571	0.1384±0.0614
water_quality	**0.8994±0.0301**	0.9034±0.0452	0.9544±0.0487	0.9087±0.0461	0.9125±0.0503
enron	**0.1872±0.0342**	0.1933±0.0302	0.1925±0.0227	0.1932±0.0161	0.1932±0.0220
image	0.1665±0.0207	**0.1616±0.0159**	0.1758±0.0135	0.4195±0.0165	0.4230±0.0231
scene	**0.0652±0.0089**	0.0659±0.0129	0.0717±0.0076	0.4156±0.0155	0.4180±0.0167
yeast	**0.4452±0.0188**	0.4455±0.0155	0.4520±0.0157	0.4844±0.0089	0.4844±0.0158
corel5k	**0.1550±0.0120**	0.1613±0.0681	0.2952±0.0160	0.3459±0.0076	0.3421±0.0101
bibtex	0.2189±0.0086	0.2173±0.0126	**0.1934±0.0154**	0.4863±0.0119	0.5016±0.0093

Table 6 Experimental results of threshold variation (average ± standard deviation) in terms of hamming loss (smaller values indicate better performance)

Data sets	$mean(W) - 0.1$	$mean(W) - 0.05$	$mean(W)$	$mean(W) + 0.05$	$mean(W) + 0.1$
flag	0.3135±0.0428	0.3116±0.0521	**0.3067±0.0493**	0.3607±0.0216	0.3453±0.0361
CAL500	0.1377±0.0049	0.1374±0.0056	0.1375±0.0032	0.1373±0.0027	**0.1372±0.0052**
emotions	0.3128±0.0186	0.3184±0.0314	0.3151±0.0163	**0.3114±0.0185**	0.3115±0.0157
genbase	**0.0011±0.0012**	0.0013±0.0010	0.0074±0.0023	0.0512±0.0034	0.0512±0.0031
medical	0.0075±0.0004	**0.0074±0.0008**	0.0236±0.0011	0.0251±0.0051	0.0239±0.0029
water_quality	**0.2985±0.0104**	0.3371±0.0150	0.3417±0.0197	0.3367±0.0176	0.3925±0.0116
enron	0.1023±0.0237	**0.1003±0.0297**	0.1004±0.0286	**0.1003±0.0292**	0.1006±0.0276
image	**0.1464±0.0122**	0.1480±0.0131	0.1592±0.0123	0.2472±0.0045	0.2472±0.0075
scene	0.0753±0.0057	**0.0747±0.0116**	0.0836±0.0074	0.1790±0.0028	0.1790±0.0030
yeast	**0.1861±0.0116**	0.1862±0.0096	0.1914±0.0111	0.2318±0.0075	0.2318±0.0070
corel5k	**0.0096±0.0000**	0.0097±0.0000	0.0108±0.0011	0.0103±0.0002	0.0101±0.0000
bibtex	**0.0115±0.0002**	**0.0115±0.0003**	0.0199±0.0003	0.0151±0.0005	0.0150±0.0005

Table 7 Experimental results of threshold variation (average ± standard deviation) in terms of one error (smaller values indicate better performance)

Data sets	$mean(W) - 0.1$	$mean(W) - 0.05$	$mean(W)$	$mean(W) + 0.05$	$mean(W) + 0.1$
flag	0.2345±0.0928	0.2196±0.0667	0.2138±0.0752	0.2568±0.1565	**0.2127±0.0911**
CAL500	0.1196±0.0378	0.1198±0.0701	0.1196±0.0453	**0.1155±0.0511**	0.1156±0.0479
emotions	0.5582±0.0403	0.5616±0.0779	**0.5479±0.0722**	0.5550±0.0848	0.5546±0.0673
genbase	0.0015±0.0048	**0.0005±0.0001**	0.0045±0.0073	0.7190±0.0391	0.7219±0.0589
medical	**0.0218±0.0035**	0.0234±0.0037	0.0227±0.0035	0.0230±0.0042	0.0277±0.0085
water_quality	0.2270±0.0513	**0.2167±0.0488**	0.3178±0.0439	0.3281±0.0414	0.3247±0.0457
enron	0.8819±0.0229	0.8654±0.0521	0.8672±0.0553	0.8644±0.0540	**0.8638±0.0746**
image	0.2530±0.0361	**0.2450±0.0311**	0.2825±0.0307	0.7100±0.0235	0.7230±0.0521
scene	**0.1824±0.0226**	0.1845±0.0373	0.2110±0.0217	0.7786±0.0203	0.7786±0.0196
yeast	**0.2114±0.0322**	0.2168±0.0256	0.2222±0.0336	0.2487±0.0349	0.2487±0.0226
corel5k	**0.5316±0.0545**	0.5528±0.0625	0.7524±0.1243	0.8208±0.0858	0.7802±0.0239
bibtex	0.3241±0.0094	**0.3231±0.0145**	0.3295±0.0180	0.8591±0.0155	0.8493±0.0154

Table 8 Experimental results of threshold variation (average ± standard deviation) in terms of ranking loss (smaller values indicate better performance)

Data sets	$mean(W) - 0.1$	$mean(W) - 0.05$	$mean(W)$	$mean(W) + 0.05$	$mean(W) + 0.1$
flag	0.2286±0.0480	0.2282±0.0413	**0.2269±0.0568**	0.2440±0.0388	0.2362±0.0412
CAL500	**0.1818±0.0059**	**0.1818±0.0056**	0.1825±0.0050	0.1819±0.0050	0.1819±0.0084
emotions	0.4232±0.0261	0.4185±0.0405	**0.4139±0.0223**	0.4293±0.0210	0.4219±0.0377
genbase	**0.0008±0.0018**	0.0009±0.0024	0.0028±0.0017	0.1657±0.0188	0.1652±0.0172
medical	**0.0012±0.0000**	0.0013±0.0001	0.0018±0.0002	0.0273±0.0014	0.0296±0.0085
water_quality	0.2783±0.0208	0.2725±0.0139	**0.2531±0.0183**	0.3017±0.0157	0.3114±0.0139
enron	**0.8238±0.0622**	0.8588±0.0552	0.8580±0.0572	0.8564±0.0565	0.8598±0.0562
image	0.1404±0.0212	**0.1343±0.0142**	0.1508±0.0167	0.4590±0.0210	0.4739±0.0272
scene	**0.0616±0.0094**	0.0621±0.0155	0.0693±0.0078	0.4799±0.0222	0.4826±0.0181
yeast	0.1572±0.0132	**0.1569±0.0142**	0.1652±0.0142	0.2086±0.0081	0.2088±0.0204
corel5k	**0.0629±0.0047**	0.0649±0.0026	0.1222±0.0067	0.1581±0.052l	0.1559±0.0048
bibtex	0.1118±0.0065	0.1080±0.0078	**0.0984±0.0067**	0.3282±0.0090	0.3300±0.0090

Table 9 Experimental results of comparing approaches (mean ± std. deviation) in terms of average precision (larger values indicate better performance)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	0.8194±0.0577	0.8285±0.0738	0.8103±0.0430	0.8001±0.0028	0.9113±0.0611	**0.9374±0.0176**
CAL500	0.5186±0.0471	**0.5251±0.0015**	0.5077±0.0750	0.4931±0.0336	0.5093±0.0951	0.4567±0.0270
emotions	0.8336±0.0396	**0.8449±0.0498**	0.8269±0.0419	0.7914±0.0431	0.7679±0.0487	0.8163±0.0341
genbase	**0.9972±0.0078**	0.9874±0.0078	0.9879±0.0073	0.9746±0.0024	0.0768±0.0091	0.9244±0.0095
medical	**0.9571±0.0194**	0.9542±0.0177	0.9296±0.0174	0.1346±0.0053	0.2101±0.0085	0.9175±0.0258
water_quality	0.5516±0.0447	**0.6782±0.0529**	0.5379±0.0492	0.2714±0.0105	0.5109±0.0317	0.6513±0.0464
enron	**0.8607±0.0285**	0.6591±0.0472	0.6912±0.0315	0.6131±0.0606	0.4072±0.0517	0.6294±0.0237
image	**0.8387±0.0198**	0.7804±0.0285	0.8235±0.0217	0.7988±0.0203	0.7602±0.0174	0.8322±0.0175
scene	**0.8900±0.0098**	0.8633±0.0051	0.8844±0.0077	0.8180±0.0136	0.8346±0.0279	0.8695±0.0263
yeast	**0.7840±0.0198**	0.7835±0.0154	0.7769±0.0173	0.7520±0.0205	0.7311±0.0225	0.7663±0.0138
corel5k	**0.2838±0.0093**	0.2232±0.0010	0.2054±0.0126	0.2456±0.0074	0.1945±0.0342	0.2015±0.0429
bibtex	**0.5766±0.0141**	0.5490±0.0063	0.5438±0.0170	0.5117±0.0241	0.4293±0.0139	0.5446±0.0060

Table 10 Experimental results of comparing approaches (mean ± std. deviation) in terms of coverage (smaller values indicate better performance)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	**0.5000±0.0356**	0.5714±0.0235	0.5391±0.0233	0.6100±0.0490	0.5302±0.0413	0.5173±0.0166
CAL500	**0.7128±0.0745**	0.7130±0.0784	0.7220±0.0538	0.7469±0.1053	0.7445±0.0984	0.7190±0.0173
emotions	0.2744±0.0178	0.2599±0.0166	0.2521±0.0175	0.3518±0.0169	0.3475±0.0167	**0.2101±0.0217**
genbase	**0.0118±0.0230**	0.0284±0.0103	0.0152±0.0338	0.0339±0.0157	0.7443±0.0687	0.0135±0.0071
medical	0.1286±0.0673	**0.0137±0.0057**	0.1195±0.0591	0.1496±0.0195	0.1356±0.0283	0.0240±0.0043
water_quality	0.9059±0.0415	**0.6354±0.0277**	0.8922±0.0320	0.6464±0.0221	0.6031±0.0273	0.6381±0.0259
enron	**0.1929±0.0672**	0.2430±0.1202	0.2202±0.0703	0.2881±0.1540	0.2875±0.1102	0.3371±0.0312
image	**0.1585±0.0066**	0.2330±0.0025	0.1720±0.0102	0.2009±0.0094	0.2170±0.0099	0.2128±0.0183
scene	**0.0650±0.0045**	0.0795±0.0022	0.0679±0.0045	0.1351±0.0076	0.0931±0.0060	0.0777±0.0019
yeast	**0.4370±0.0234**	0.4430±0.0211	0.4477±0.0252	0.4537±0.0238	0.4715±0.2446	0.4605±0.0229
corel5k	**0.2905±0.0931**	0.3014±0.0176	0.3166±0.0904	0.3227±0.1742	0.7590±0.1435	0.3514±0.0269
bibtex	0.1904±0.0134	**0.0728±0.0106**	0.2089±0.0100	0.2710±0.0852	0.2342±0.0674	0.0780±0.0051

Table 11 Experimental results of comparing approaches (mean ± std. deviation) in terms of hamming loss (smaller values indicate better performance)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	**0.2594±0.0049**	0.2782±0.0012	0.2714±0.0048	0.2917±0.0137	0.6214±0.0128	0.2872±0.0195
CAL500	0.1368±0.0015	**0.1347±0.0003**	0.1370±0.0011	0.9671±0.0022	0.9684±0.0021	0.2105±0.0170
emotions	0.1784±0.0265	0.1638±0.0289	0.1826±0.0312	0.2994±0.0716	0.8475±0.0288	**0.1446±0.0274**
genbase	**0.0070±0.0007**	0.0103±0.0004	0.0120±0.0011	0.0285±0.0012	0.0900±0.0023	0.0146±0.0042
medical	**0.0074±0.0008**	0.0087±0.0008	0.0077±0.0008	0.5172±0.0654	0.9697±0.0569	0.0093±0.0002
water_quality	**0.2983±0.0144**	0.2985±0.0082	0.2898±0.0090	0.8445±0.0083	0.8524±0.0193	0.6387±0.0597
enron	**0.0106±0.0035**	0.0629±0.0015	0.0350±0.0032	0.0870±0.0036	0.1279±0.0112	0.0337±0.0031
image	0.1489±0.0105	0.8750±0.0056	0.1551±0.0092	0.1956±0.0167	0.2070±0.0158	**0.1282±0.0184**
scene	0.0728±0.0061	**0.0574±0.0049**	0.0784±0.0052	0.0985±0.0075	0.8875±0.0068	0.0793±0.0094
yeast	**0.1834±0.0098**	0.8058±0.0078	0.1874±0.0118	0.1991±0.0127	0.2267±0.0103	0.1956±0.0111
corel5k	**0.0106±0.0013**	0.0109±0.0002	0.0179±0.0025	0.0257±0.0019	0.7108±0.0010	0.0253±0.0031
bibtex	**0.0117±0.0063**	0.0120±0.0010	0.0175±0.0023	0.1506±0.0072	0.0288±0.0021	0.0782±0.0059

Table 12 Experimental results of comparing approaches (mean ± std. deviation) in terms of one error (smaller values indicate better performance)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	0.2127±0.02724	0.1579±0.0894	0.2288±0.1015	0.2448±0.0836	**0.0526±0.0682**	0.1946±0.0635
CAL500	**0.1153±0.0282**	0.6000±0.0210	0.1163±0.0235	0.1196±0.0274	0.2000±0.0548	0.1557±0.0098
emotions	**0.2034±0.0602**	0.2125±0.0519	0.2123±0.0673	0.2813±0.0751	0.3220±0.0823	0.2091±0.0207
genbase	**0.0015±0.0047**	0.0555±0.0372	0.0061±0.0085	0.0612±0.0418	0.8911±0.0016	0.0592±0.0212
medical	**0.0217±0.0027**	0.0612±0.0149	0.0267±0.0031	0.2395±0.0285	0.1633±0.0116	0.0633±0.0099
water_quality	**0.2159±0.0469**	0.2547±0.0273	0.2868±0.0330	0.3009±0.0357	0.3208±0.0648	0.2371±0.0261
enron	0.8564±0.0481	0.1560±0.0417	0.2977±0.0435	0.6537±0.0328	0.6048±0.0346	**0.1523±0.0154**
image	**0.2490±0.0331**	0.2850±0.0251	0.2675±0.0237	0.2900±0.0391	0.3700±0.0390	0.2510±0.0304
scene	**0.1849±0.0152**	0.1853±0.0032	0.1949±0.0175	0.2400±0.0264	0.2833±0.0247	0.1872±0.0293
yeast	**0.2077±0.0171**	0.2438±0.0217	0.2977±0.0156	0.6537±0.0284	0.7048±0.0237	0.2095±0.0123
corel5k	**0.7102±0.0165**	0.8874±0.0138	0.7305±0.0137	0.8133±0.0102	0.9817±0.0260	0.8156±0.0188
bibtex	**0.3216±0.0251**	0.3327±0.0243	0.3544±0.0255	0.3971±0.0198	0.5746±0.0160	0.3301±0.0186

Table 13 Experimental results of comparing approaches (mean ± std. deviation) in terms of ranking loss (smaller values indicate better performance)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	0.2058±0.0295	0.1860±0.0394	0.2123±0.0427	0.2392±0.0333	**0.1061±0.0319**	0.1817±0.0281
CAL500	**0.1714±0.0109**	0.1734±0.0074	0.1790±0.0071	0.1829±0.0099	0.1738±0.0122	0.2102±0.0174
emotions	0.1350±0.0299	**0.1128±0.0295**	0.1392±0.0281	0.1596±0.0394	0.2010±0.0279	0.1442±0.0138
genbase	**0.0025±0.0072**	0.0100±0.0073	0.0120±0.0071	0.0200±0.0065	0.0099±0.0018	0.0098±0.0014
medical	**0.0013±0.0001**	0.0059±0.0002	0.0021±0.0002	0.0194±0.0024	0.0130±0.0037	0.0067±0.0018
water_quality	**0.2501±0.0194**	0.2525±0.0220	0.2619±0.0178	0.2721±0.0231	0.2695±0.0217	0.2802±0.0219
enron	**0.0891±0.0108**	0.0943±0.0092	0.0911±0.0194	0.1038±0.0266	0.2135±0.0163	0.1348±0.0164
image	0.1306±0.0149	0.2033±0.0132	0.1465±0.0172	0.1871±0.0202	0.1950±0.0177	**0.1287±0.0187**
scene	**0.0611±0.0066**	0.0720±0.0074	0.0646±0.0075	0.0937±0.0100	0.0928±0.0108	0.0712±0.0081
yeast	**0.1503±0.0154**	0.1569±0.0147	0.1572±0.0138	0.1731±0.0144	0.1835±0.0140	0.1652±0.0102
corel5k	**0.1188±0.0038**	0.2015±0.0061	0.1476±0.0038	0.3192±0.0072	0.9991±0.0070	0.2543±0.0157
bibtex	0.0989±0.0072	**0.0733±0.0079**	0.1128±0.0056	0.0818±0.0048	0.2301±0.0091	0.0789±0.0217

metrics of average precision, hamming loss and one error at τ=mean(W)-0.05, achieves the optimal performance with respect to two evaluation metrics of coverage and ranking loss at τ=mean(W). We choose to achieve superior parameter τ on more evaluation metrics, so we make the parameter τ=mean(W)-0.05 in the following comparative experiments.

4.3.2 Performance experiment

Tables 9, 10, 11, 12 and 13 report the detailed experimental results on five multi-label evaluation metrics respectively. For each evaluation metric, the maximum values denoting the best performance are in bold. It is obvious that in most of the cases, our method provides the best performance. For example, concerning the metric of one error, our method ranks first on 83% (10/12) data sets (Table 12). By comparing the performance of the proposed method Relief-LIFT with the most popular multi-label learning methods (MLNB, FRS-SS-LIFT, BILAS, k-MAR and MLFE), it is found that the proposed method achieves satisfactory prediction results on the vast majority of multi-label data sets, implying that label-specific feature selection can greatly improve the learning performance of multi-label learning systems. As can be seen from Table 14, on 11 datasets, we can see that the Relief-LIFT algorithm consumes less time than the other five algorithms while maintaining superior performance. Furthermore, we use the widely-used Friedman test [44] to perform statistical comparisons of multiple methods across multiple data sets. Table 15 reports Friedman statistics for all evaluation metrics along with cutoffs at the 0.05 significance level. As shown in Table 15, the F_F value is greater than the critical value of 2.3828 on all evaluation indicators. Therefore, the null hypothesis of "equal" performance between comparison methods should be explicitly rejected. The Bonferroni-Dunn test [44] is used as a post-hoc test to show the relative performance between the comparison methods. Here, the difference between the mean rank of the control method (i.e. Relief-LIFT) and one of the comparison methods is calibrated with the critical difference (CD). The performances of Relief-LIFT and one of the comparison methods are considered significantly different if their mean rankings differed by at least one CD (CD=2.1767). The mean ranking for each comparison algorithm is marked along the axis as shown in Fig. 1, the mean ranking of Relief-LIFT on the five evaluation metrics is better than all comparison algorithms, and it is at least one CD away from the average ranking of MLNB and k-MAR, which indicates that Relief-LIFT significantly outperforms the two comparative performances of the algorithms.

Table 15 Friedman statistics F_F in terms of each evaluation metric as well as the critical value at 0.05 significance level (comparing methods $n = 6$, data sets $N = 12$)

Evaluation metric	F_F	critical value
Average precision	12.7329	
Coverage	6.5000	
Hamming loss	22.5593	2.3828
One error	11.3910	
Ranking loss	8.4118	

Table 16 lists the comparison of label-specific feature dimensions after feature selection with the original label-specific feature dimensions. We can clearly see that by feature selection, the number of features in Relief-LIFT is greatly reduced compared with the original label-specific feature space. For example, on bibtex multi-label data set,

Table 14 Time consumptions of each comparing algorithm on the 12 data sets (seconds)

Data sets	Relief-LIFT	BILAS	FRS-SS-LIFT	k-MAR	MLNB	MLFE
flag	**0.2093**	0.8271	3.0529	0.3491	6.8891	0.8743
CAL500	**15.0508**	424.7864	226.3472	252.2269	466.8025	213.5716
emotions	**2.0285**	2.4480	30.3678	16.4625	30.7047	2.3598
genbase	**2.9357**	50.6217	6.9449	14267.3462	101.9531	13.9671
medical	**16.5273**	252.3208	76.0711	20139.3598	369.8942	53.1437
water_quality	29.7362	32.8619	168.4970	**7.5657**	50.3855	61.8534
enron	**27.8178**	53.2870	87.9902	13085.9147	337.9281	49.7168
image	**56.0748**	60.9024	2073.5934	6416.8176	108.8350	89.6605
scene	**85.2341**	140.6824	1771.8048	9454.2476	147.8665	277.0812
yeast	**260.7478**	273.2318	21376.5367	1371.5031	247.4068	320.4934
corel5k	**3562.3587**	4391.2674	73091.1942	17541.0543	10576.9778	4651.5138
bibtex	**7695.5979**	10035.0489	120587.4983	56193.0074	12273.7845	12015.2496

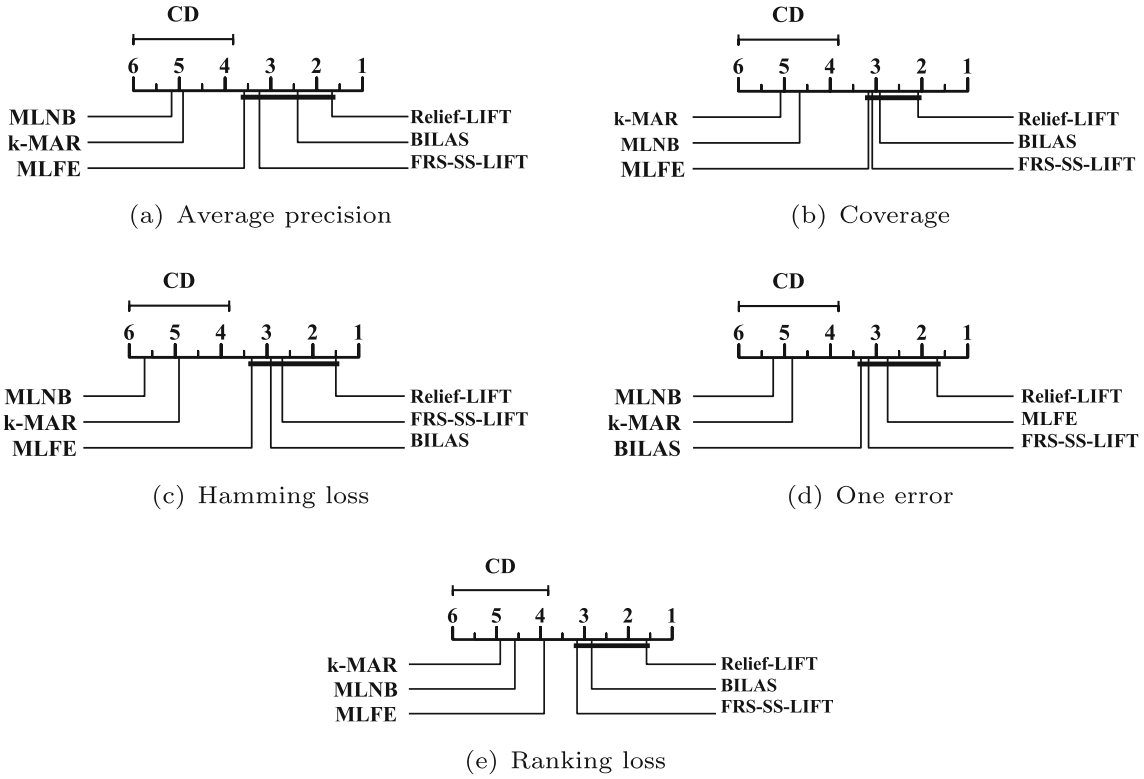

Fig. 1 Comparison of Relief-LIFT (control method) against four comparing methods with the Bonferroni-Dunn test under each evaluation criterion. Relief-LIFT is superior to the other five comparison algorithms in every evaluation index, and it is significantly superior to k-MAR and MLNB algorithms. Methods not connected with Relief-LIFT in the CD diagram are considered to have significantly different performance from the control method (CD=2.1767 at 0.05 significance level)

the average number of features per label after building the label-specific feature space is 45.58. Fortunately, Relief-LIFT selects an average of 17.50 relevant features per label, and it has an advantage in the performance comparison with the five algorithms. Correspondingly, we have reason to believe that LIFT easily causes redundancy or irrelevance in tailored feature spaces that degrades the learning performance, and Relief-LIFT can alleviate such problem. These results clearly demonstrate the superior effectiveness of our method against other well-established multi-label learning algorithms.

5 Conclusion

The effectiveness of LIFT strategy has been extensively demonstrated for multi-label learning. However, the construction of label-specific features may lead to an increase in the dimension of features with redundant information. To

Table 16 Label-specific feature dimensions on 12 data sets in $LIFT_k$ and $Relief-LIFT_k$

	Feature space			Feature space	
Data	$LIFT_k$	$Relief\text{-}LIFT_k$	Data	$LIFT_k$	$Relief\text{-}LIFT_k$
flag	26.57	10.71	enron	12.55	3.20
CAL500	27.56	13.19	image	198.00	85.20
emotions	74.67	37.17	scene	173.33	61.00
genbase	14.75	5.71	yeast	225.00	69.86
medical	11.82	11.10	corel5k	16.61	8.00
water_quality	59.42	52.86	bibtex	45.58	17.50

this end, a new method named Relief-LIFT which is integrated by LIFT and a modified Relief was proposed in this paper. On the one hand, LIFT was leveraged to generate the tailored features which reveal high-level information. On the other hand, Relief was improved to select the identified features which capture relevant and necessary information. Furthermore, experimental studies showed that Relief-LIFT outperformed four other compared algorithms in both time efficiency and classification effectiveness.

It is worth noting that Relief-LIFT does not fully consider the correlation between different labels. To further improve the performance of our multi-label learning method, it is a useful attempt to explore the correlation between binary tree classifiers and labels. In addition, the real data contained in multi-label learning is not comprehensive, and multi-view learning is also a direction worth studying.

Acknowledgements This work is supported by the National Natural Science Foundation of China (62076111, 62006128, 62006099).

Data Availability The image data generated or analysed during this study is included in this published article [Refer: M.L. Zhang, Z.H. Zhou, ML-kNN: A lazy learning approach to multi-label learning, Pattern Recognition, 40, 2038-2048 (2007)]. The remaining data sets generated during or analysed during the current study are available in the Mulan repository, [http://mulan.sourceforge.net/index.html].

References

1. Sun S, Zong D (2020) Lcbm: a multi-view probabilistic model for multi-label classification. IEEE T Pattern Anal 43(8):2682–2696
2. Law A, Ghosh A (2021) Multi-label classification using binary tree of classifiers. IEEE T Em Top Comp I 6(3):677–689
3. Wever M, Tornede A, Mohr F, Hüllermerier E (2021) AutoML for multi-label classification: overview and empirical evaluation. IEEE T Pattern Anal 43(9):3037–3054
4. Chen Z, Ren J (2021) Multi-label text classification with latent word-wise label information. Appl Intell 51:966–979
5. Zhang P, Liu G, Gao W, Song J (2021) Multi-label feature selection considering label supplementation. Pattern Recogn 120:108137
6. Pereira RB, Plastino A, Zadrozny B, Merschmann LH (2018) Categorizing feature selection methods for multi-label classification. Artif Intell Rev 49(1):57–78
7. Cevikalp H, Benligiray B, Gerek ON (2020) Semi-supervised robust deep neural networks for multi-label image classification. Pattern Recogn 100:107164
8. Liu T, Wang J, Yang B, Wang X (2021) Facial expression recognition method with multi-label distribution learning for non-verbal behavior understanding in the classroom. Infrared Phys Techn 112:103594
9. Liu T, Yang B, Liu H, Ju J, Tang J, Subramanian S, Zhang Z (2022) GMDL: toward Precise head pose estimation via Gaussian mixed distribution learning for students' attention understanding. Infrared Phys Techn 122:104099
10. Liu T, Wang J, Yang B, Wang X (2021) NGDNEt: nonuniform Gaussian-label distribution learning for infrared head pose estimation and on-task behavior understanding in the classroom. Neurocomputing 436:210–220
11. Fernandes MS, Cordeiro W, Recamonde-Mendoza M (2021) Detecting Aedes aegypti mosquitoes through audio classification with convolutional neural networks. Comput Biol Med 129:104152
12. Koutini K, Eghbal-zadeh H, Widmer G (2021) Receptive field regularization techniques for audio classification and tagging with deep convolutional neural networks. IEEE-ACM T Audio SPE 29:1987–2000
13. Shrestha R, Glackin C, Wall J, Cannings N (2021) Bird audio diarization with faster R-CNN. In: International conference on artificial neural networks. Springer, Cham, pp 415–426
14. Liu H, Wang X, Zhang W, Zhang Z, Li YF (2020) Infrared head pose estimation with multi-scales feature fusion on the IRHP database for human attention recognition. Neurocomputing 411:510–520
15. Liu H, Zheng C, Li D, Zhang Z, Lin K, Shen X et al (2022) Multi-perspective social recommendation method with graph representation learning. Neurocomputing 468:469–481
16. Liu H, Nie H, Zhang Z, Li YF (2021) Anisotropic angle distribution learning for head pose estimation and attention understanding in human-computer interaction. Neurocomputing 433:310–322
17. Zhang ML, Wu L (2014) LIFT: multi-Label learning with label-specific features. IEEE T Pattern Anal 37(1):107–120
18. Sun S, Zong D (2020) Lcbm: a multi-view probabilistic model for multi-label classification. IEEE T Pattern Anal 43(8):2682–2696
19. Xu SP, Yang XB, Yu HL, Yu DJ, Yang J, Tsang EC (2016) Multi-label learning with label-specific feature reduction. Knowl-Based Syst 104:52–61
20. Kira K, Rendell LA (1992) A practical approach to feature selection. In: 9th International conference on machine learning. Morgan Kaufmann, pp 249-256
21. Tsoumakas G, Katakis I, Vlahavas I (2009) Mining multi-label data. Data Min Knowl Disc 2009:667–685
22. Liu T, Li YF, Liu H, Zhang Z, Liu S (2019) RISIR: rapid infrared spectral imaging restoration model for industrial material detection in intelligent video systems. IEEE T Ind Inform, Early Access. https://doi.org/10.1109/TII.2019.2930463
23. Liu T, Liu H, Li YF, Chen Z, Zhang Z, Liu S (2019) Flexible FTIR spectral imaging enhancement for industrial robot infrared vision sensing. IEEE T Ind Inform 16(1):544–554
24. Liu T, Liu H, Li Y, Zhang Z, Liu S (2018) Efficient blind signal reconstruction with wavelet transforms regularization for educational robot infrared vision sensing. IEEE-Asme T Mech 24(1):384–394
25. Boutell MR, Luo J, Shen X, Brown CM (2004) Learning multi-label scene classification. Pattern Recogn 37(9):1757–1771
26. Zhang ML, Zhou ZH (2007) ML-KNN: a lazy learning approach to multi-label learning. Pattern Recogn 40(7):2038–2048
27. Zhang ML, Peña JM, Robles V (2009) Feature selection for multi-label naive Bayes classification. Inform Sciences 179:3218–3229
28. Liu H, Liu T, Zhang Z, Sangaiah AK, Yang B, Li Y (2022) ARHPE: asymmetric relation-aware representation learning for head pose estimation in industrial human ccomputer interaction. IEEE T Ind Inform 18(10):7107–7117
29. Liu H, Zheng C, Li D, Shen X et al (2021) EDMF: efficient deep matrix factorization with review feature learning for industrial recommender system. IEEE T Ind Inform 18(7):4361–4371
30. Zhang JJ, Fang M, Li X (2015) Multi-label learning with discriminative features for each label. Neurocomputing 154:305–316
31. Huang J, Li G, Huang Q, Wu X (2015) Learning label specific features for multi-label classification. In: 2015 IEEE international conference on data mining. IEEE, pp 181–190

32. Zhang Y, Gong DW, Sun XY, Guo YN (2017) A PSO-based multi-objective multi-label feature selection method in classification. Sci Rep-UK 7(1):1–12
33. Zhang J, Li C, Cao D, Lin Y, Su S, Dai L, Li S (2018) Multi-label learning with label-specific features by resolving label correlations. Knowl-Based Syst 159:148–157
34. Li H, Li DY, Wang SG (2015) Multi-label learning with label-specific features based on rough sets. J Comput Syst Sci 36(12):2730–2734
35. Sinaga KP, Yang MS (2020) Unsupervised K-means clustering algorithm. IEEE Access 8:80716–80727
36. Zhang J, Luo Z, Li C, Zhou C, Li S (2019) Manifold regularized discriminative feature selection for multi-label learning. Pattern Recogn 95:136–150
37. Chen T, Lin L, Hui X, Chen R, Wu H (2020) Knowledge-guided multi-label few-shot learning for general image recognition. IEEE T Pattern Anal. https://ieeexplore.ieee.org/document/9207855/
38. Zhang Y, Wang Y, Liu XY, Mi S, Zhang ML (2020) Large-scale multi-label classification using unknown streaming images. Pattern Recogn 99:107100
39. Zhang ML, Fang JP (2020) Partial multi-label learning via credible label elicitation. IEEE T Pattern Anal 43(10):3587–3599
40. Wang J, Yang L, Huo Z, He W, Luo J (2020) Multi-label classification of fundus images with efficientnet. IEEE Access 8:212499–212508
41. Fan X, Chen X, Wang C, Wang Y, Zhang Y (2022) Margin attribute reductions for multi-label classification. Appl Intell 52(6):6079–6092
42. Zhang ML, Fang JP, Wang YB (2021) Bilabel-specific features for multi-label classification. ACM T Knowl Discov D 16(1):1–23
43. Zhang QW, Zhong Y, Zhang ML (2018) Feature-induced labeling information enrichment for multi-label learning. In: Proceedings of the 32nd AAAI conference on artificial intelligence (AAAI'18). New Orleans, TX, 4446-4453
44. Yu ZB, Zhang ML (2021) Multi-label classification with label-specific feature generation: a wrapped approach. IEEE T Pattern Anal 44(9):5199–5210

Affiliations

Jiadong Zhang[1] · Keyu Liu[1,2] · Xibei Yang[1,3] · Hengrong Ju[4] · Suping Xu[5,6]

Jiadong Zhang
zh18257375003@163.com

Xibei Yang
jsjxy_yxb@just.edu.cn

Hengrong Ju
juhengrong@ntu.edu.cn

Suping Xu
supingxu@yahoo.com

1 School of Computer, Jiangsu University of Science and Technology, Zhenjiang, 212100, Jiangsu, China

2 School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, 611756, Sichuan, China

3 Key Laboratory of Oceanographic Big Data Mining & Application of Zhejiang Province, Zhejiang Ocean University, Zhoushan, 316022, Zhejiang, China

4 School of Information Science and Technology, Nantong University, Nantong, 226000, Jiangsu, China

5 Department of Computer Science and Technology, Nanjing University, Nanjing, 210023, Jiangsu, China

6 Department of Electrical and Computer Engineering, University of Alberta, Edmonton, T6R 2V4, AB, Canada