Knowledge-Based Systems 104 (2016) 52–61

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Multi-label learning with label-specific feature reduction

Suping Xu[a,e,f], Xibei Yang[a,b,e,f,*], Hualong Yu[a], Dong-Jun Yu[c], Jingyu Yang[c], Eric C.C. Tsang[d]

[a] School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, PR China
[b] School of Economics and Management, Nanjing University of Science and Technology, Nanjing 210094, PR China
[c] Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information, Nanjing University of Science and Technology, Ministry of Education, Nanjing 210094, PR China
[d] Faculty of Information Technology, Macau University of Science and Technology, 519020, Macau
[e] Intelligent Information Processing Key Laboratory of Shanxi Province, Shanxi University, Taiyuan 030006, PR China
[f] Key Laboratory of Oceanographic Big Data Mining and Application of Zhejiang Province, Zhejiang Ocean University, Zhoushan 316022, PR China

ARTICLE INFO

Article history:
Received 22 December 2015
Revised 10 March 2016
Accepted 13 April 2016
Available online 25 April 2016

Keywords:
Feature reduction
Fuzzy rough set
Label-specific feature
Multi-label learning
Sample selection

ABSTRACT

In multi-label learning, since different labels may have some distinct characteristics of their own, multi-label learning approach with label-specific features named LIFT has been proposed. However, the construction of label-specific features may encounter the increasing of feature dimensionalities and a large amount of redundant information exists in feature space. To alleviate this problem, a multi-label learning approach FRS-LIFT is proposed, which can implement label-specific feature reduction with fuzzy rough set. Furthermore, with the idea of sample selection, another multi-label learning approach FRS-SS-LIFT is also presented, which effectively reduces the computational complexity in label-specific feature reduction. Experimental results on 10 real-world multi-label data sets show that, our methods can not only reduce the dimensionality of label-specific features when compared with LIFT, but also achieve satisfactory performance among some popular multi-label learning approaches.

1. Introduction

Nowadays, multi-label learning problem has received an increased attention in real-world applications. For example, in semantic annotation of images [3,16,26,49], a picture can be annotated as camel, desert and landscape. In text categorization [5,11,17,29], a document may belong to several given topics, including economics, finance or GDP. In bioinformatics [6,13,50], each gene may be associated with a set of functional classes, such as metabolism, transcription and protein synthesis. In all cases above, each sample may be associated with more than one label simultaneously and predefined labels for different samples are not mutually exclusive but may overlap. This situation is distinct from the traditional single-label learning where predefined labels are mutually exclusive, each sample only belongs to a single label.

Over the last decade, many multi-label learning approaches have been witnessed [12,28,58]. Generally, the existing methods can be grouped into two main categories [43], i.e., algorithm adaptation methods and problem transformation methods. Algorithm adaptation methods extend specific single-label learning algorithms to directly handle multi-label data by modifying some constraint conditions, such as AdaBoost.MH [40], ML-*k*NN [59], MLNB [60], and RankSVM [9]. Problem transformation methods, transform the multi-label task into one or more corresponding single-label ones and then handle them one by one through traditional methods. The well-known problem transformation methods include binary relevance (BR), label power set (LP) and pruned problem transformation (PPT). BR [3] learns a binary classifier for each label independently and predicts each of the labels separately, so it cuts up the relationship among different labels. LP [44] considers each unique set of labels that exists in a multi-label training set as a new single-label multi-value class. Though this method considers the correlations among different labels, it easily leads to a higher time consumption since the number of new classes is increased exponentially with the increasing of labels. Meanwhile, some new classes created by a few samples may lead to class unbalance problem. PPT [34] abandons the new classes associated with extremely small number of samples or assigns these samples with new labels that can create accepted classes, while some abandoned classes will lead to the loss of multi-label information. Although above methods have achieved good performance in multi-label learning, they make use of the same features to achieve

[*] Corresponding author at: School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang 212003, PR China.

E-mail addresses: supingxu@yahoo.com (S. Xu), yangxibei@hotmail.com (X. Yang), yuhualong@just.edu.cn (H. Yu), njyudj@njust.edu.cn (D.-J. Yu), yangjy@mail.njust.edu.cn (J. Yang), cctsang@must.edu.mo (E.C.C. Tsang).

http://dx.doi.org/10.1016/j.knosys.2016.04.012
0950-7051/© 2016 Elsevier B.V. All rights reserved.

the learning purposes in different labels. Actually, different labels may have distinct characteristics of their own, and these characteristics are more inclined to judge whether labels belong to a specific sample. Fortunately, Zhang [61,62] has proposed the representative LIFT algorithm and validated the effectiveness of constructing label-specific features. For each label, LIFT employs clustering analysis in the positive and negative samples respectively, and then constructs label-specific features by checking the distances between the sample and all the clustering centers. (There is not any semanteme for constructed label-specific features, which can be regarded as a set of distances.) However, construction of label-specific features may encounter the increasing of feature dimensionalities, and a large amount of redundant information exists in feature space. As a result, the structure information between different samples will be disrupted, and even more be destroyed, which leads to the decreasing of the performance of multi-label learning approach. To alleviate this problem, an effective solution is to perform dimension reduction in label-specific features.

Rough set theory is a good mathematical tool for describing incomplete and uncertain data. With over 30 years of development, it has been widely applied in attribute reduction [18,30], feature selection [20,22,31,42,55], rule extraction [25,38] and uncertainty reasoning [46]. Numerous researchers [31,32] have used the various rough set models for dealing with single-label data analyses in real-world applications. Recently, some researchers [53,54,56,57] begin to attempt at carrying out multi-label classification via rough set approaches, however, all of them determine different labels in the same feature space, which contradicts the fact that different labels may have distinct characteristics of their own. In this paper, with the idea of attribute reduction based on fuzzy rough set, we will develop a multi-label learning approach with label-specific feature reduction (FRS-LIFT), which uses the approximation quality to evaluate the significance of specific dimension and takes the forward greedy search strategy. Furthermore, sample selection is an effective data compression technique, which can reduce the time and memory consumption in attribute reduction. On the basis of FRS-LIFT, another multi-label learning approach with label-specific feature reduction by sample selection (FRS-SS-LIFT) will be presented at the same time. To validate the effectiveness of FRS-LIFT and FRS-SS-LIFT, we conduct comprehensive experiments on 10 real-world multi-label data sets. Experimental study shows clear advantages of FRS-LIFT and FRS-SS-LIFT over various multi-label learning algorithms.

The rest of this paper is organized as following. Section 2 introduces the formal definition of multi-label learning's framework and LIFT approach. Section 3 provides some background materials on fuzzy rough set and sample selection, and then the details of our FRS-LIFT and FRS-SS-LIFT are presented. Section 4 describes data sets, evaluation metrics, experimental settings, and then analyzes the results of comparative studies on 10 multi-label data sets. Finally, Section 5 summarizes and sets up several issues for future work.

2. Multi-label learning

2.1. Multi-label learning's framework

Let $X = \mathbb{R}^d$ be the d-dimensional sample space and $L = \{l_1, l_2, \ldots, l_m\}$ be the finite set of m possible labels. $T = \{(x_i, Y_i) | i = 1, 2, \ldots, n\}$ denotes the multi-label training set with n labeled samples, where $x_i \in X$ is a d-dimensional feature vector such that $x_i = [x_i^1, x_i^2, \ldots, x_i^d]$, $Y_i \subseteq L$ is the set of labels associated with x_i.

The goal of multi-label learning is to produce a real-valued function $f : X \times P(L) \to \mathbb{R}$. In detail, for each $x_i \in X$, a prefect learning system will tend to output larger values for labels in Y_i than those not in Y_i [59], i.e., for any l, $l' \in L$, if $l \in Y_i$ and $l' \notin Y_i$, $f(x_i, l) > f(x_i, l')$ holds.

2.2. LIFT approach

2.2.1. Construction for label-specific features

LIFT aims to improve the learning performance of multi-label learning system through generating distinguishing features which capture the specific characteristics of each label $l_k \in L$. To achieve this goal, LIFT takes into account intrinsic connection between different samples in all labels. 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:

$$P_k = \left\{x_i \middle| (x_i, Y_i) \in T, l_k \in Y_i\right\}; \tag{1}$$

$$N_k = \left\{x_i \middle| (x_i, Y_i) \in T, l_k \notin Y_i\right\}. \tag{2}$$

In other words, the training sample x_i belongs to P_k if x_i has label l_k; otherwise, x_i is included in N_k.

To consider intrinsic connection among different samples, LIFT employs clustering analysis on P_k and N_k, respectively. Following Zhang's research [61,62], k-means algorithm [21] is adopted to partition P_k into m_k^+ disjoint clusters whose clustering centers are denoted by $\{p_1^k, p_2^k, \ldots, p_{m_k^+}^k\}$. Similarly, N_k is also partitioned into m_k^- disjoint clusters whose clustering centers are $\{n_1^k, n_2^k, \ldots, n_{m_k^-}^k\}$. LIFT treats clustering information gained from P_k and N_k as equal importance, and then the numbers of clusters on P_k and N_k are set to be the same, i.e., $m_k^+ = m_k^- = m_k$. Specifically, the number of clusters for both positive samples and negative samples is:

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

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 inner structures of positive samples P_k and negative samples N_k, on this basis, label-specific features can be constructed in the form of:

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

where $d(\cdot, \cdot)$ represents the distance between two samples. In literatures [61,62], Euclidean metric is used to calculate sample distance. Actually, φ_k is a mapping from the original d-dimensional sample space X to a new $2m_k$-dimensional label-specific feature space $LIFT_k$, i.e., φ_k: $X \to LIFT_k$.

2.2.2. Induction for classification models

LIFT induces a family of m classification models $\{f_1, f_2, \ldots, f_m\}$ in the constructed label-specific feature spaces $LIFT_k (1 \le k \le m)$. Formally, for each $l_k \in L$, a binary training set T_k^* with n samples is created from the training set T according to the mapping φ_k, such that:

$$T_k^* = \left\{(\varphi_k(x_i), \phi(Y_i, l_k)) \middle| (x_i, Y_i) \in T\right\}, \tag{5}$$

where $\phi(Y_i, l_k) = +1$ if $l_k \in Y_i$; otherwise, $\phi(Y_i, l_k) = -1$. Based on the binary training set T_k^*, any binary learner can be employed to induce a classification model f_k: $LIFT_k \to \mathbb{R}$ for l_k.

Given an unseen sample $x' \in X$, the predicted label set for x' is $Y' = \{l_k | f(\varphi_k(x'), l_k) > 0, 1 \le k \le m\}$.

3. Multi-label learning with label-specific feature reduction

3.1. Fuzzy rough set

To fuse rough set approaches into machine learning problems, we will introduce the classification learning task instead of the notion of information system. Formally, a classification learning task can also be considered as the 3-tuple $< U, A, D >$, in which $U = \{x_1, x_2, \ldots, x_n\}$ is the finite set of n samples called the universe of discourse, $A = \{a_1, a_2, \ldots, a_c\}$ is the set of condition features, D is the decision.

Let $U \neq \emptyset$ be a universe of discourse. F: $U \to [0, 1]$ is a fuzzy set [8] on U, $F(x)$ is the membership function of F, $F(U)$ is the set of all fuzzy sets on U. A given fuzzy binary relation R can be a fuzzy equivalence relation if and only if R is reflexive, symmetric and transitive. Equivalently, $\forall x,\ y,\ z \in U$, $R(x, x) = 1$, $R(x, y) = R(y, x)$ and $\bigwedge_y (R(x, y), R(y, z)) \leq R(x, z)$.

Definition 1 ([8,18]). Let $U \neq \emptyset$ be a universe of discourse, R is a fuzzy equivalence relation on U, $\forall F \in F(U)$, the fuzzy lower and upper approximations of F are denoted by $\underline{R}\,(F)$ and $\overline{R}(F)$, respectively, $\forall x \in U$, the membership functions are defined as:

$$\underline{R}(F)(x) = \inf_{y \in U} \max\left(1 - R(x, y), F(y)\right); \tag{6}$$

$$\overline{R}(F)(x) = \sup_{y \in U} \min\left(R(x, y), F(y)\right). \tag{7}$$

The pair $[\underline{R}(F), \overline{R}(F)]$ is referred to as a fuzzy rough set of F.

Definition 2. Let $< U, A, D >$ be a classification learning task, $\forall B \subseteq A$, R_B is the fuzzy equivalence relation on U in feature subset B, $U/IND(D) = \{d_1, d_2, \ldots, d_p\}$ is the partition induced by the decision D, then approximate quality of $U/IND(D)$ based on fuzzy rough set is represented in form of:

$$\gamma(B, D) = \frac{\left|\bigcup_{i=1}^{p} \underline{R_B}(d_i)\right|}{|U|} = \frac{\sum_{j=1}^{|U|} \left(\bigvee_{i=1}^{p} \underline{R_B}(d_i)(x_j)\right)}{|U|}, \tag{8}$$

where $|\cdot|$ denotes the cardinality of a set, d_i is a decision class.

$\gamma(B,\ D)$ reflects the approximation abilities of the granulated space induced by feature subset B to characterize the decision D. Obviously, $0 \leq \gamma(B,\ D) \leq 1$ holds. In literatures [18,19], it is proved that approximate quality is monotonic with the increasing or decreasing of condition features in a classification learning task, i.e, $\gamma(B_1,\ D) \leq \gamma(B_2,\ D)$ if $B_1 \subseteq B_2$.

Definition 3. Let $< U, A, D >$ be a classification learning task, $\forall B \subseteq A$, B is referred to as a reduct of A if and only if

1. $\gamma(B, D) = \gamma(A, D)$;
2. $\forall C \subset B$, $\gamma(C,\ D) \neq \gamma(B,\ D)$.

By Definition 3, we can see that a reduct of A is a minimal subset of A, which preserves the approximate quality. However, in the majority of real-world applications, the above definition is much too strict. To expand the application scope of attribute reduction (dimension reduction, feature selection), Hu et al. [18,19] introduced the threshold ε to control the change of approximate quality for loosening the restrictions of reduct. In reality, we can also consider B as a reduct of A when satisfying the following conditions: (1) $\gamma(A, D) - \gamma(B, D) \leq \varepsilon$; (2) $\forall C \subset B$, $\gamma(A, D) - \gamma(C, D) > \varepsilon$. Note that, ε is aimed at reducing redundant information as much as possible, while maintaining the change of approximate quality in a smaller range. In general, ε is recommended to be [0, 0.1].

In fuzzy rough set, with the number of features increasing, the fuzzy similarity between samples will decrease, and then the lower approximation of decision will increase, namely, the size of positive region will be enlarged. As is well known, the samples in positive region are usually regarded as to be certain, and then the degree of certainty in the classification learning task will be improved. It is consistent with our intuition that new features will bring new information about granulation and classification.

Let $< U, A, D >$ be a classification learning task, $\forall a_i \in B \subseteq A$, we define a coefficient

$$Sig_{in}(a_i, B, D) = \gamma(B, D) - \gamma(B - \{a_i\}, D) \tag{9}$$

as the significance of a_i in B relative to decision D. $Sig_{in}(a_i, B, D)$ reflects the change of approximate quality if a_i is eliminated from B. Accordingly, we can also define

$$Sig_{out}(a_i, B, D) = \gamma(B + \{a_i\}, D) - \gamma(B, D), \tag{10}$$

where $a_i \in A - B$, $Sig_{out}(a_i, B, D)$ measures the change of approximate quality if a_i is introduced into B. On the basis of above, large numbers of researchers [24,45,47,51,52,63] iteratively select the most significant features with forward greedy algorithm until no more deterministic rules generating with the increasing of features. Feature selection (dimension reduction) based on approximate quality can greatly reduce redundant and irrelevant information in feature space, while remaining the degree of certainty in the classification learning task.

3.2. Label-specific feature reduction approach

In this subsection, we will propose a multi-label learning approach with label-specific feature reduction based on fuzzy rough set (FRS-LIFT). In the multi-label training set T, FRS-LIFT firstly constructs the label-specific feature space $LIFT_k$ for each label l_k (Steps 2–4); then, dimension reduction in $LIFT_k$ is implemented with fuzzy rough set (Steps 5–10); next, m classification models are built in the dimension-reduced label-specific feature space $FRS\text{-}LIFT_k$ (Steps 13 and 14); finally, the unseen sample is predicted in the multi-label learned system (Step 16).

Formally, FRS-LIFT can be designed as following.

The time complexity of FRS-LIFT mainly comprises of three components: clustering on P_k and N_k in Step 3, forming the label-specific feature space in Step 4, and dimension reduction for the label-specific feature space in Steps 5–10. The cost of performing clustering on P_k and N_k using k-means is $O(m_k(t_1|P_k| + t_2|N_k|))$, 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 requires $O(2m_k|T|)$ time. Finally, the time complexity of dimension reduction is $O(4m_k^2|T|^2)$. Therefore, in general the time complexity of FRS-LIFT is $O(m_k(t_1|P_k| + t_2|N_k|) + 2m_k|T| + 4m_k^2|T|^2)$.

Although FRS-LIFT improves the performance of multi-label learning via reducting redundant label-specific feature dimensionalities, its computational complexity is high. To alleviate this problem, sample selection technique will be introduced in next subsection.

3.3. Sample selection

In the field of machine learning, sample selection is considered as a better data compression technique [23]. The ultimate goal of sample selection is to reduce the size of training samples without losing any extractable information, while simultaneously insisting that a learning approach built on the reduced training samples is good or nearly as good as a learning approach built on the original training samples [7]. It is obvious that removing some samples from the training set decreases the computational complexity of learning approach. Several methods of sample selection have been explored and studied, such as condensed nearest neighbor (CNN) algorithm [15], instance-based learning (IB) algorithm [1], selective nearest neighbor (SNN) algorithm [36], and edited nearest neighbor (ENN) algorithm [48].

According to many research [4,48], the uncertainty for samples in the boundary is larger than in other places, which means that the information provided by boundary samples will be more important. Accordingly, the majority of methods of sample selection

Algorithm 1 Multi-label learning algorithm with label-specific feature reduction.

```
Inputs: The multi-label training set $T$, the ratio parameter $\delta$ for controlling the number of clusters, the threshold $\varepsilon$ for controlling the change of approximate quality, the unseen sample $x'$;
Outputs: The predicted label set $Y'$.
1.  For $l_k$ from $l_1$ to $l_m$ do
2.      Form the set of positive samples $P_k$ and the set of negative samples $N_k$ based on $T$ according to Eqs. (1) and (2);
3.      Perform $k$-means clustering on $P_k$ and $N_k$, each with $m_k$ clusters as defined in Eq. (3);
4.      $\forall(x_i, Y_i) \in T$, create the mapping $\varphi_k(x_i)$ according to Eq. (4), form the original label-specific feature space $LIFT_k$ for label $l_k$;
5.      Compute $\gamma(A, l_k)$, where $A$ is the set of label-specific features in $LIFT_k$;
6.      $B \leftarrow \emptyset$;
7.      $\forall a_i \in A$, compute $Sig_{in}(a_i, A, l_k)$;
8.      $B \leftarrow a_j$, where $Sig_{in}(a_j, A, l_k) = \max\{Sig_{in}(a_i, A, l_k)\colon \forall a_i \in A\}$, compute $\gamma(B, l_k)$;
9.      Do
          1) $\forall a_i \in A - B$, compute $Sig_{out}(a_i, B, l_k)$;
          2) $B \leftarrow B \cup \{a_j\}$, where $Sig_{out}(a_j, B, l_k) = \max\{Sig_{out}(a_i, B, l_k)\colon \forall a_i \in A - B\}$;
          3) Compute $\gamma(B, l_k)$;
        Until $\gamma(A, l_k) - \gamma(B, l_k) \le \varepsilon$;
10.     $\forall a_i \in B$, if $\gamma(A, l_k) - \gamma(B - \{a_i\}, l_k) \le \varepsilon$, then $B \leftarrow B - \{a_i\}$, form the dimension-reduced label-specific
        feature space $FRS\text{-}LIFT_k$ for label $l_k$, i.e., the mapping $\varphi'_k(x_i)$;
11. End for
12. For $l_k$ from $l_1$ to $l_m$ do
13.     Construct the binary training set $T^*_k$ in $\varphi'_k(x_i)$ according to Eq. (5);
14.     Induce the classification model $f_k$: $FRS\text{-}LIFT_k \to \mathbb{R}$ by invoking any binary learner on $T^*_k$;
15. End for
16. The predicted label set $Y' = \{l_k | f(\varphi'_k(x'), l_k) > 0, 1 \le k \le m\}$.
```

tend to choose samples in boundary. Similarly, in dimension reduction, we compute a series of similarity matrices in the sample space constructed by boundary samples instead of the original sample space. The time and memory consumption of constructing fuzzy equivalence relations will be reduced greatly. For this purpose, some clustering algorithms, for example, k-means or fuzzy k-means [10,27] algorithm can be employed to seek the samples far away from the center of similar samples. Specifically, we suppose that C_j is a cluster, C^*_j is the clustering center of C_j, $dist(x, C^*_j)$ denotes the distance between $x \in C_j$ and C^*_j, and the average distance between $\forall x \in C_j$ and C^*_j is represented as following:

$$\overline{dist}(C^*_j) = \frac{1}{\langle C_j \rangle} \sum_{x \in C_j} dist(x, C^*_j), \tag{11}$$

where $\langle \cdot \rangle$ represents the number of samples in a cluster.

The sample $x \in C_j$ whose $dist(x, C^*_j)$ is larger than $\overline{dist}(C^*_j)$ is considered as a boundary sample. With all selected boundary samples, a new classification learning task can be constructed and its computational complexity is reduced in some extent.

3.4. Label-specific feature reduction approach with sample selection

In this subsection, we will propose a multi-label learning approach with label-specific feature reduction by sample selection (FRS-SS-LIFT). In the multi-label training set T, FRS-SS-LIFT firstly constructs the label-specific feature space $LIFT_k$ for each label l_k (Steps 2–4); then, sample selection is adopted to reduce the number of samples in $LIFT_k$ (Steps 5–8); next, dimension reduction in the sample-selected label-specific feature space $SS\text{-}LIFT_k$ is implemented with fuzzy rough set (Steps 9–14); after that, m classification models are built in the dimension-reduced label-specific feature space $FRS\text{-}SS\text{-}LIFT_k$ (Steps 17 and 18); finally, the unseen sample is predicted in the multi-label learned system (Step 20).

Note that k-means algorithm [21] is used to partition all samples into k clusters, where k represents the number of decision classes. In multi-label learning's framework, for each label $l \in L$, the value of decision for any $x_i \in X$ equals $+1$ if x_i has l; otherwise, the value equals -1. Therefore, k is set to be 2 in k-means clustering. The samples far away from their own clustering centers are selected to form a new label-specific feature space.

Formally, FRS-SS-LIFT can be designed as following.

The time complexity of FRS-SS-LIFT mainly comprises of four components: clustering on P_k and N_k in Step 3, forming the label-specific feature space in Step 4, sample selection on the label-specific feature space in Steps 5–8, and dimension reduction for the sample-selected label-specific feature space in Steps 9–14. The cost of performing clustering on P_k and N_k using k-means is $O(m_k(t_1|P_k| + t_2|N_k|))$, 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 requires $O(2m_k|T|)$ time. Then, sample selection on the label-specific feature space needs $O(2t_3|T|)$ time, where t_3 is the iterations of k-means on $|T|$. Finally, the time complexity of dimension reduction is $O(4m_k^2|T_s|^2)$, where $|T_s|$ is the number of selected samples (boundary samples). Therefore, in general the time complexity of FRS-SS-LIFT is $O(m_k(t_1|P_k| + t_2|N_k|) + 2m_k|T| + 2t_3|T| + 4m_k^2|T_s|^2)$.

In the majority of data sets, we have $|T| - |T_s| > t_3/(2m_k^2)$, then $(1 + |T_s|/|T|)(|T| - |T_s|) > t_3/(2m_k^2) \Leftrightarrow 2m_k^2(|T| + |T_s|)(|T| - |T_s|) > t_3|T| \Leftrightarrow 4m_k^2|T|^2 > 2t_3|T| + 4m_k^2|T_s|^2$ holds. Therefore, it is shown that the time complexity of FRS-SS-LIFT is lower than that of FRS-LIFT.

4. Experimental analysis

4.1. Datasets

To evaluate the performances of our multi-label learning methods, 10 real-world multi-label data sets have been employed in this paper. For each multi-label data set $S = \{(x_i, Y_i) | 1 \le i \le p\}$, symbol $|S|$, $dim(S)$, $L(S)$ and $F(S)$ represent the number of samples, number of features, number of possible labels, and feature type, respectively. Moreover, for better describing the characteristics of data

Algorithm 2 Multi-label learning algorithm with label-specific feature reduction by sample selection.

```
Inputs: The multi-label training set $T$, the ratio parameter $\delta$ for controlling the number of clusters, the threshold $\varepsilon$ for controlling the change of approximate quality, the unseen sample $x'$;
Outputs: The predicted label set $Y'$.
For $l_k$ from $l_1$ to $l_m$ do
    Form the set of positive samples $P_k$ and the set of negative samples $N_k$ based on $T$ according to Eqs. (1) and (2);
    Perform $k$-means clustering on $P_k$ and $N_k$, each with $m_k$ clusters as defined in Eq. (3);
    $\forall (x_i, Y_i) \in T$, create the mapping $\varphi_k(x_i)$ according to Eq. (4), form the original label-specific feature space $LIFT_k$ for label $l_k$;
    Perform $k$-means clustering in $LIFT_k$ with 2 clusters, i.e., $C_1, C_2$;
    $SS_1 \leftarrow \emptyset$, $SS_2 \leftarrow \emptyset$;
    For $C_j$ from $C_1$ to $C_2$ do
        1) $\forall x \in C_j$, compute $dist(x, C_j^*)$;
        2) Compute the average distance $\overline{dist}(C_j^*)$;
        3) $\forall x \in C_j$, if $dist(x, C_j^*) > \overline{dist}(C_j^*)$, then $SS_j \leftarrow SS_j \cup \{x\}$;
    End for
    $SS \leftarrow SS_1 \cup SS_2$, form the label-specific feature space with sample selection $SS\text{-}LIFT_k$ for label $l_k$;
    Compute $\gamma(A, l_k)$, where $A$ is the set of label-specific features in $SS\text{-}LIFT_k$;10. $B \leftarrow \emptyset$;
    $\forall a_i \in A$, compute $Sig_{in}(a_i, A, l_k)$;
    $B \leftarrow a_j$, where $Sig_{in}(a_j, A, l_k) = \max\{Sig_{in}(a_i, A, l_k)\colon \forall a_i \in A\}$, compute $\gamma(B, l_k)$;
    Do
        1) $\forall a_i \in A - B$, compute $Sig_{out}(a_i, B, l_k)$;
        2) $B \leftarrow B \cup \{a_j\}$, where $Sig_{out}(a_j, B, l_k) = \max\{Sig_{out}(a_i, B, l_k)\colon \forall a_i \in A - B\}$;
        3) Compute $\gamma(B, l_k)$;
    Until $\gamma(A, l_k) - \gamma(B, l_k) \le \varepsilon$;
    $\forall a_i \in B$, if $\gamma(A, l_k) - \gamma(B - \{a_i\}, l_k) \le \varepsilon$, then $B \leftarrow B - \{a_i\}$, form the dimension-reduced label-specific feature space $FRS\text{-}SS\text{-}LIFT_k$ for label $l_k$, i.e., the mapping $\varphi'_k(x_i)$;
End for
For $l_k$ from $l_1$ to $l_m$ do
    Construct the binary training set $T_k^*$ in $\varphi'_k(x_i)$ according to Eq. (5);
    Induce the classification model $f_k$: $FRS\text{-}SS\text{-}LIFT_k \to \mathbb{R}$ by invoking any binary learner on $T_k^*$;
End for
The predicted label set $Y' = \{l_k | f(\varphi'_k(x'), l_k) > 0, 1 \le k \le m\}$.
```

Table 1
Characteristics of the experimental data sets.

Data sets	$\|S\|$	$dim(S)$	$L(S)$	$F(S)$	$LCard(S)$	$LDen(S)$	$DL(S)$	$PDL(S)$	Domain
Emotions [2]	593	72	6	Numeric	1.869	0.311	27	0.046	Music
Birds [2]	645	260	19	Both	1.014	0.053	133	0.206	Audio
Genbase [2]	662	1186	27	Nominal	1.252	0.046	32	0.048	Biology
Medical [2]	978	1449	45	Nominal	1.245	0.028	94	0.096	Text
Enron [59]	1702	1001	53	Nominal	4.275	0.081	704	0.414	Text
Image [59]	2000	294	5	Numeric	1.236	0.247	20	0.010	Image
Scene [2]	2407	294	6	Numeric	1.074	0.179	15	0.006	Image
Yeast [2]	2417	103	14	Numeric	4.237	0.303	198	0.082	Biology
Slashdot [35]	3782	1079	22	Nominal	0.901	0.041	99	0.026	Text
Yahoo [2]	5000	640	21	Numeric	1.420	0.068	232	0.046	Text

sets, some other multi-label properties [33,58,61,62] also have been adopted such as:

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

Table 1 summarizes some detailed statistics of multi-label data sets used in our experiments. The 10 data sets are chosen from five distinct practical application domains, such as music, audio, biology, text and image. Therefore, the multi-label data sets used in our experiments are more comprehensive.

4.2. Configuration

Since each sample is simultaneously associated with several labels, the performance evaluation in multi-label learning is more complicated than traditional single-label learning. Some popular evaluation metrics in the single-label learning system, such as accuracy, precision, recall and F-measure [41], can't well adapt to the multi-label learning system. In this paper, five widely used multi-label evaluation metrics proposed in [14,37,39,40] are employed, including average precision, coverage, hamming loss, one error and ranking loss.

Given a multi-label testing set $T' = \{(x_i, Y_i)|1 \le i \le t\}$, the real-valued function $f(\cdot, \cdot)$ produced from the multi-label learning system can be transformed into a ranking function $rank_f(\cdot, \cdot)$ [59]. 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. The detailed multi-label evaluation metrics are presented as following.

Table 2
Predictive performance of each comparing algorithm (mean ± std. deviation) on Emotions.

Emotions	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.8030 ± 0.0498	1.7797 ± 0.2100	0.1916 ± 0.0268	0.2662 ± 0.0760	0.1585 ± 0.0393
MLNB	0.7999 ± 0.0419	1.8190 ± 0.1680	0.2056 ± 0.0286	0.2678 ± 0.0813	0.1635 ± 0.0279
LIFT	0.8251 ± 0.0396	1.6922 ± 0.1768	**0.1786** ± **0.0295**	0.2223 ± 0.0581	0.1408 ± 0.0297
FRS-LIFT	**0.8280** ± **0.0411**	1.6940 ± 0.1873	0.1798 ± 0.0290	**0.2155** ± **0.0608**	**0.1401** ± **0.0299**
FRS-SS-LIFT	0.8268 ± 0.0400	**1.6890** ± **0.1714**	0.1809 ± 0.0310	0.2223 ± 0.0651	0.1406 ± 0.0280

Table 3
Predictive performance of each comparing algorithm (mean ± std. deviation) on Birds.

Birds	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.5926 ± 0.0611	**2.0469** ± **0.3535**	0.0473 ± 0.0051	0.4779 ± 0.0751	0.1619 ± 0.0336
MLNB	0.5443 ± 0.0430	2.5203 ± 0.4002	0.0857 ± 0.0120	0.5250 ± 0.0675	0.1965 ± 0.0319
LIFT	0.6102 ± 0.0577	2.3249 ± 0.2090	0.0432 ± 0.0038	0.4139 ± 0.0937	0.1750 ± 0.0286
FRS-LIFT	0.6449 ± 0.0738	2.1662 ± 0.3484	**0.0411** ± **0.0048**	0.3855 ± 0.1061	0.1615 ± 0.0392
FRS-SS-LIFT	**0.6528** ± **0.0763**	2.0678 ± 0.4198	0.0421 ± 0.0031	**0.3792** ± **0.1064**	**0.1558** ± **0.0423**

- Average Precision [37]: evaluates the average fraction of labels ranked above a particular label $l \in Y_i$ which actually are in Y_i. The bigger the value of *AveragePrecision(f)*, the better the performance. Specially, the performance is perfect when $AveragePrecision(f) = 1$.

$$AveragePrecision(f) = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i|} \times \sum_{l \in Y_i}\frac{\left|\left\{l' \middle| rank_f(x_i, l') \le rank_f(x_i, l), l' \in Y_i\right\}\right|}{rank_f(x_i, l)}. \tag{12}$$

- Coverage [40]: evaluates the average depth of going down the list of labels for covering all the possible labels of sample. The smaller the value of *Coverage(f)*, the better the performance. Specially, the performance is perfect when $Coverage(f) = 0$.

$$Coverage(f) = \frac{1}{t}\sum_{i=1}^{t}\max_{l \in Y_i} rank_f(x_i, l) - 1. \tag{13}$$

- Hamming Loss [14]: evaluates the average time of misclassification in each sample, i.e., a label not associated with the sample is predicted or a label associated with the sample is not predicted. The smaller the value of *HammingLoss(h)*, the better the performance. Specially, the performance is perfect when $HammingLoss(h) = 0$.

$$HammingLoss(h) = \frac{1}{t}\sum_{i=1}^{t}\left|h(x_i) \otimes Y_i\right|, \tag{14}$$

where $h(x_i)$ is the predicted label set which is associated with x_i, $\otimes$ represents symmetric difference between two sets.

- One Error [39]: evaluates the average fraction of top-ranked label which is not in the set of possible labels associated with the sample. The smaller the value of *OneError(f)*, the better the performance. Specially, the performance is perfect when $OneError(f) = 0$.

$$OneError(f) = \frac{1}{t}\sum_{i=1}^{t}\Psi\left(\left[\arg\max_{l \in L} f(x_i, l)\right] \notin Y_i\right), \tag{15}$$

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

- Ranking Loss [59]: evaluates the average fraction of label pairs which are reversely ordered for the sample. The smaller the value of *RankingLoss(f)*, the better the performance. Specially, the performance is perfect when $RankingLoss(f) = 0$.

$$RankingLoss(f) = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i||\overline{Y_i}|}\left|\left\{(l, l') \middle| f(x_i, l) \le f(x_i, l'), (l, l') \in Y_i \times \overline{Y_i}\right\}\right|, \tag{16}$$

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

In this paper, our FRS-LIFT and FRS-SS-LIFT are compared with three well-established multi-label learning algorithms, including ML-*k*NN [59], MLNB [60] and LIFT [62]. According to Ref. [59], in ML-*k*NN, the number of nearest neighbors k and smoothing parameter s are set to be 10 and 1, respectively. For LIFT, FRS-LIFT and FRS-SS-LIFT, we adjust the parameter δ by increasing it from 0.1 to 1.0 (stepsize 0.1), and finally assign δ to 0.2. Note that, to our best knowledge, no theoretical bases have been reported to specify the threshold ε for controlling the change of approximate quality. The optimal value of threshold ε is dependent on the nature of a specific application. Therefore, we conduct a large number of experiments which help us determine an optimal change range of approximate quality. Consequently, our methods achieve better classification performance when ε is between 0.001 and 0.05. Furthermore, all experiments are run on a workstation equipped with a 3.10 Hz processor and a 8.00G memory.

4.3. Results

10-fold cross-validation (10-CV) is used for evaluating the effectiveness of different methods in our experiments. 10-CV breaks all samples into 10 groups of the same size, the nine groups compose the multi-label training set and the one group composes the multi-label testing set. The classification process repeats 10 times in turn and the mean value and standard deviation of 10 experimental results are recorded.

Table 2–11 demonstrate the performance comparisons of our methods with some other multi-label learning methods on above 10 data sets respectively. For each evaluation metric, ↑ indicates the larger the better while ↓ indicates the smaller the better. Meanwhile, the best performance among the five comparing algorithms is highlighted in boldface.

In all the 50 predictive performance results (10 data sets × 5 evaluation metrics), FRS-LIFT ranks in first place among the five comparing algorithms at 44% cases, in second place at 30% cases, in third place at 26% cases, and never ranks in fourth and fifth places. Meanwhile, FRS-SS-LIFT ranks in first place at 42% cases, in second place at 48% cases, in third place at 8% cases, and only 2%

Table 4
Predictive performance of each comparing algorithm (mean ± std. deviation) on Genbase.

Genbase	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.9851 ± 0.0123	0.5653 ± 0.2834	0.0045 ± 0.0022	0.0151 ± 0.0213	0.0068 ± 0.0064
MLNB	0.0599 ± 0.0044	21.0660 ± 0.6873	0.0464 ± 0.0020	1.0000 ± 0.0000	**0.0000 ± 0.0000**
LIFT	0.9935 ± 0.0078	0.4733 ± 0.3308	0.0023 ± 0.0015	**0.0015 ± 0.0047**	0.0048 ± 0.0072
FRS-LIFT	**0.9944 ± 0.0078**	**0.4415 ± 0.3172**	**0.0015 ± 0.0009**	**0.0015 ± 0.0047**	0.0043 ± 0.0071
FRS-SS-LIFT	0.9935 ± 0.0085	0.4684 ± 0.3380	0.0017 ± 0.0011	0.0030 ± 0.0094	0.0051 ± 0.0077

Table 5
Predictive performance of each comparing algorithm (mean ± std. deviation) on Medical.

Medical	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.8068 ± 0.0248	2.6177 ± 0.7451	0.0158 ± 0.0015	0.2546 ± 0.0262	0.0397 ± 0.0093
MLNB	0.0866 ± 0.0056	13.321 ± 1.0635	0.0304 ± 0.0040	0.3313 ± 0.0517	0.0430 ± 0.0122
LIFT	0.8831 ± 0.0145	1.7502 ± 0.6276	0.0114 ± 0.0013	0.1564 ± 0.0258	0.0246 ± 0.0072
FRS-LIFT	**0.9096 ± 0.0176**	1.7293 ± 0.7365	**0.0087 ± 0.0014**	**0.1124 ± 0.0279**	0.0248 ± 0.0108
FRS-SS-LIFT	0.9087 ± 0.0155	**1.6897 ± 0.5035**	0.0089 ± 0.0013	0.1186 ± 0.0231	**0.0236 ± 0.0074**

Table 6
Predictive performance of each comparing algorithm (mean ± std. deviation) on Enron.

Enron	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.5134 ± 0.0327	13.3890 ± 1.2027	0.0482 ± 0.0043	0.5158 ± 0.0417	0.1638 ± 0.0222
MLNB	0.3679 ± 0.0212	16.9268 ± 1.1020	0.1147 ± 0.0115	0.6248 ± 0.0345	0.2448 ± 0.0163
LIFT	0.5620 ± 0.0321	12.2104 ± 1.2205	0.0365 ± 0.0034	0.4279 ± 0.0456	0.1352 ± 0.0190
FRS-LIFT	**0.6611 ± 0.0408**	**10.8708 ± 0.9921**	**0.0341 ± 0.0032**	**0.3084 ± 0.0444**	**0.0953 ± 0.0107**
FRS-SS-LIFT	0.6481 ± 0.0287	11.1791 ± 0.7025	0.0372 ± 0.0034	0.3256 ± 0.0437	0.1046 ± 0.0099

Table 7
Predictive performance of each comparing algorithm (mean ± std. deviation) on Image.

Image	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.7900 ± 0.0203	0.9780 ± 0.1034	0.1701 ± 0.0141	0.3195 ± 0.0332	0.1765 ± 0.0202
MLNB	0.7578 ± 0.0217	1.0855 ± 0.0991	0.1951 ± 0.0143	0.3730 ± 0.0390	0.2012 ± 0.0177
LIFT	0.8337 ± 0.0153	0.8085 ± 0.0778	0.1524 ± 0.0123	0.2535 ± 0.0270	0.1349 ± 0.0132
FRS-LIFT	0.8314 ± 0.0177	0.8245 ± 0.0811	0.1479 ± 0.0103	0.2555 ± 0.0334	0.1378 ± 0.0149
FRS-SS-LIFT	**0.8364 ± 0.0162**	**0.7995 ± 0.1015**	**0.1468 ± 0.0097**	**0.2490 ± 0.0226**	**0.1323 ± 0.0171**

Table 8
Predictive performance of each comparing algorithm (mean ± std. deviation) on Scene.

Scene	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.8687 ± 0.0164	0.4694 ± 0.0573	0.0852 ± 0.0060	0.2185 ± 0.0313	0.0760 ± 0.0100
MLNB	0.8436 ± 0.0155	0.5276 ± 0.0599	0.1065 ± 0.0067	0.2651 ± 0.0247	0.0886 ± 0.0106
LIFT	0.8909 ± 0.0090	**0.3801 ± 0.0448**	0.0757 ± 0.0070	0.1865 ± 0.0161	0.0593 ± 0.0071
FRS-LIFT	0.8913 ± 0.0084	0.3855 ± 0.0374	**0.0740 ± 0.0052**	0.1841 ± 0.0156	0.0601 ± 0.0061
FRS-SS-LIFT	**0.8921 ± 0.0101**	0.3809 ± 0.0452	0.0751 ± 0.0057	**0.1836 ± 0.0195**	**0.0592 ± 0.0072**

Table 9
Predictive performance of each comparing algorithm (mean ± std. deviation) on Yeast.

Yeast	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.7659 ± 0.0194	6.2679 ± 0.2381	0.1934 ± 0.0116	0.2251 ± 0.0284	0.1666 ± 0.0149
MLNB	0.7478 ± 0.0148	6.4989 ± 0.2443	0.2094 ± 0.0100	0.2445 ± 0.0237	0.1772 ± 0.0140
LIFT	0.7731 ± 0.0178	6.2559 ± 0.2221	0.1900 ± 0.0107	0.2189 ± 0.0217	0.1614 ± 0.0141
FRS-LIFT	0.7762 ± 0.0172	6.2609 ± 0.2570	0.1875 ± 0.0114	0.2147 ± 0.0171	0.1588 ± 0.0150
FRS-SS-LIFT	**0.7790 ± 0.0167**	**6.2249 ± 0.2529**	**0.1869 ± 0.0111**	**0.2085 ± 0.0156**	**0.1560 ± 0.0138**

Table 10
Predictive performance of each comparing algorithm (mean ± std. deviation) on Slashdot.

Slashdot	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.8835 ± 0.0116	1.0144 ± 0.1745	0.0221 ± 0.0010	0.0946 ± 0.0143	0.0497 ± 0.0072
MLNB	0.8381 ± 0.0167	1.0555 ± 0.1423	0.0378 ± 0.0027	0.1777 ± 0.0262	0.0538 ± 0.0070
LIFT	0.8927 ± 0.0091	0.8516 ± 0.1653	**0.0159 ± 0.0009**	0.0898 ± 0.0134	0.0418 ± 0.0062
FRS-LIFT	**0.9045 ± 0.0098**	**0.5449 ± 0.0811**	**0.0159 ± 0.0011**	**0.0858 ± 0.0162**	**0.0289 ± 0.0038**
FRS-SS-LIFT	0.9038 ± 0.0074	0.6044 ± 0.0940	0.0160 ± 0.0011	0.0864 ± 0.0138	0.0311 ± 0.0038

Table 11
Predictive performance of each comparing algorithm (mean ± std. deviation) on Yahoo.

Yahoo	Average precision ↑	Coverage ↓	Hamming loss ↓	One error ↓	Ranking loss ↓
ML-*k*NN	0.5428 ± 0.0102	3.4044 ± 0.0936	0.0632 ± 0.0015	0.6378 ± 0.0195	0.1271 ± 0.0041
MLNB	0.6019 ± 0.0155	3.2978 ± 0.1955	0.0702 ± 0.0031	0.5318 ± 0.0169	0.1221 ± 0.0090
LIFT	**0.6986 ± 0.0176**	**2.5990 ± 0.1997**	0.0495 ± 0.0015	0.3898 ± 0.0248	**0.0880 ± 0.0079**
FRS-LIFT	0.6940 ± 0.0183	2.7526 ± 0.1834	0.0492 ± 0.0019	0.3872 ± 0.0251	0.0939 ± 0.0078
FRS-SS-LIFT	0.6963 ± 0.0141	2.7136 ± 0.1401	**0.0491 ± 0.0019**	**0.3846 ± 0.0252**	0.0924 ± 0.0056

Table 12
Label-specific feature dimensionalities of three comparing algorithms on the 10 data sets.

Algorithm	Emotions	Birds	Genbase	Medical	Enron	Image	Scene	Yeast	Slashdot	Yahoo
LIFT	74.67	14.63	13.33	11.82	55.70	198.00	173.33	225.00	38.91	136.00
FRS-LIFT	58.50	14.47	10.85	10.27	39.25	174.40	122.67	180.90	32.45	124.62
FRS-SS-LIFT	56.83	14.16	9.07	10.09	33.94	171.80	119.67	179.90	31.77	124.29

Table 13
Numbers of selected samples of two comparing algorithms on the 10 data sets.

Algorithm	Emotions	Birds	Genbase	Medical	Enron	Image	Scene	Yeast	Slashdot	Yahoo
FRS-LIFT	593	645	662	978	1702	2000	2407	2417	3782	5000
FRS-SS-LIFT	265	240	256	422	633	829	1035	1198	1763	1822

Table 14
Time consumptions of two comparing algorithms on the 10 data sets (seconds).

Algorithm	Emotions	Birds	Genbase	Medical	Enron	Image	Scene	Yeast	Slashdot	Yahoo
FRS-LIFT	592	88	165	729	14,206	19,837	23,891	144,103	130,720	743,359
FRS-SS-LIFT	97	13	19	163	2008	3522	4442	36,239	29,121	107,099

cases exist in fourth place. From the performance comparisons of proposed methods (FRS-LIFT and FRS-SS-LIFT) with the most popular multi-label learning methods (ML-*k*NN, MLNB and LIFT), it is found that the proposed methods achieve satisfactory predictive results on the large majority of multi-label data sets, which means that label-specific feature reduction can greatly improve the learning performance of multi-label learning system.

Table 12 lists the label-specific feature dimensionalities of three comparing algorithms, including LIFT, FRS-LIFT and FRS-SS-LIFT, which construct multi-label learning systems from the view of label-specific features. On all the 10 data sets, the label-specific feature dimensionalities of our FRS-LIFT and FRS-SS-LIFT are both smaller than LIFT. Taking Yeast as an example for detailed analysis, we can find that the label-specific feature dimensionality of LIFT is 225.00, however, the label-specific feature dimensionalities of FRS-LIFT and FRS-SS-LIFT fall to 180.90 and 179.90, respectively. The decreasing of label-specific feature dimensionalities means that fuzzy rough set based dimension reduction really is an effective approach to eliminate redundant information in label-specific feature space. So naturally, our methods speed up the training process in comparison to LIFT.

Table 13 and 14 list the detailed comparison of FRS-LIFT and FRS-SS-LIFT. From Table 13, it is clear that, via sample selection, the number of samples used for dimension reduction in FRS-SS-LIFT has been greatly reduced in comparison to FRS-LIFT. Although the maximum data compression ratio reaches 50.43% (minimum is 63.56%) in our experiments, as shown in Table 2–11, the predictive performance differences between FRS-SS-LIFT and FRS-LIFT is insignificant, and both of them are superior to some other popular multi-label learning methods. Table 14 shows that time consumption of FRS-SS-LIFT is much smaller than that of FRS-LIFT. For example, on Slashdot, the number of samples used in FRS-LIFT is 3782, suppose that we construct fuzzy equivalence relations for label-specific feature reduction in such an enormous sample space, the memory cost will be very huge. Fortunately, FRS-SS-LIFT chooses 1763 representative samples from the whole 3782 samples, and then the operating efficiency of learning approach has been extremely improved. From Table 14, we can see that time consumption of FRS-LIFT on Slashdot is 130,720 s while FRS-SS-LIFT is only 29,121 s. When the available multi-label datasets are large, such as multimedia databases, genome sequences and financial forecasting, the advantage of FRS-SS-LIFT is more obvious. In some ways, FRS-SS-LIFT can be considered as an evolution of FRS-LIFT.

5. Conclusions

Different labels may have distinct characteristics of their own, and then construction of label-specific features for each label is great necessary for multi-label learning. However, the construction of label-specific features may cause the increasing of feature dimensionalities with redundant information. In this paper, we have developed two approaches named FRS-LIFT and FRS-SS-LIFT for multi-label learning, which effectively remove some redundant information existing in label-specific feature space with the idea of fuzzy rough set based attribute reduction. In addition, FRS-SS-LIFT using sample selection comes with the equivalent predictive performance while achieving the low computational complexity in comparison to FRS-LIFT. In other words, FRS-SS-LIFT can be considered as an evolution of FRS-LIFT. The experimental study on 10 data sets from five different application domains demonstrates the superiorities of our proposed two approaches to other three typical multi-label learning approaches, including ML-*k*NN, MLNB and LIFT.

It is worth noting that FRS-LIFT and FRS-SS-LIFT do not take full account of the correlations between different labels. To further improve the performances of our multi-label learning approaches, we can attempt to fuse them into dimension-reduced label-specific features in the future study.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Nos. 61572242, 61305058, 61373062, 61503160, 61502211), Key Program of National Natural Science Foundation of China (No. 61233011), Natural Science Foundation of Jiangsu Province of China (No. BK20130471), Open Project Foundation of Intelligent Information Processing Key Laboratory of Shanxi Province (No. 2014002), Key Laboratory of Oceanographic Big Data Mining & Application of Zhejiang Province (No. OBDMA201501), Postdoctoral Science Foundation of China (No. 2014M550293), Qing Lan Project of Jiangsu Province of China, Postgraduate Research Innovation Foundation of Jiangsu University of Science and Technology (No. YCX15S-10).

References

[1] D.W. Aha, D. Kibler, M.K. Albert, Instance-based learning algorithms, Mach. Learn. 6 (1991) 37–66.

[2] J. Alcala-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. Garcia, L. Sanchez, F. Herrera, KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework, J. Multiple-Valued Logic Soft Comput. 17 (2011) 255–287.

[3] M.R. Boutell, J.B. Luo, X.P. Shen, M.B. Christopher, Learning multi-label scene classification, Pattern Recognit. 37 (2004) 1757–1771.

[4] H. Brighton, C. Mellish, Advances in instance selection for instance-based learning algorithms, Data Min. Knowl. Discov. 6 (2002) 153–172.

[5] E. Chang, G. Sychay, CBSA: content-based soft annotation for multimodal image retrieval using bayes point machines, IEEE Trans. Circuits Syst. Video Technol. 13 (2003) 26–38.

[6] K.C. Chou, H.B. Shen, EUK-MPLOC: a fusion classifier for large-scale eukaryotic protein subcellular location prediction by incorporating multiple sites, J. Proteome Res. 6 (2007) 1728–1734.

[7] I. Czarnowski, Cluster-based instance selection for machine classification, Knowl. Inf. Syst. 30 (2012) 113–133.

[8] D. Dubois, H. Prade, Rough fuzzy sets and fuzzy rough sets, Int. J. Gen. Syst. 17 (1990) 191–209.

[9] A. Elisseeff, J. Weston, A kernel method for multi-labelled classification, in: Proceedings of the 14th Annual Conference on Advances in Neural Information Processing Systems, British Columbia, Canada, 2001, pp. 681–687.

[10] S. Eschrich, J.W. Ke, L.O. Hall, D.B. Goldgof, Fast accurate fuzzy clustering through data reduction, IEEE Trans. Fuzzy Syst. 11 (2003) 262–270.

[11] A. Esuli, T. Fagni, F. Sebastiani, Boosting multi-label hierarchical text categorization, Inf. Retr. 11 (2008) 287–313.

[12] E. Gibaja, S. Ventura, Multi-label learning: a review of the state of the art and ongoing research, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 4 (2014) 411–444.

[13] E. Glory, R.F. Murphy, Automated subcellular location determination and high--throughput microscopy, Dev. Cell 12 (2007) 7–16.

[14] S. Godbole, S. Sarawagi, Discriminative methods for multi-labeled classification, in: Proceedings of the Advances in Knowledge Discovery and Data Mining, 2004, pp. 22–30.

[15] P.E. Hart, The condensed nearest neighbour rule, IEEE Trans. Inf. Theory 14 (1968) 515–516.

[16] Z.Y. He, C. Chen, J.J. Bu, P. Li, D. Cai, Multi-view based multi-label propagation for image annotation, Neurocomputing 168 (2015a) 853–860.

[17] Z.Y. He, J. Wu, T. Li, Label correlation mixture model: a supervised generative approach to multilabel spoken document categorization, IEEE Trans. Emerg. Top. Comput. 3 (2015b) 235–245.

[18] Q.H. Hu, L. Zhang, D.G. Chen, W. Pedrycz, D.R. Yu, Gaussian kernel based fuzzy rough sets: model, uncertainty measures and applications, Int. J. Approx. Reason. 51 (2010a) 453–471.

[19] Q.H. Hu, W. Pedrycz, D.R. Yu, J. Lang, Selecting discrete and continuous features based on neighborhood decision error minimization, IEEE Trans. Syst. Man Cybern. Part B 40 (2010b) 137–150.

[20] A.S. Iquebal, A. Pal, D. Ceglarek, M.K. Tiwari, Enhancement of mahalanobis–taguchi system via rough sets based feature selection, Expert Syst. Appl. 41 (2014) 8003–8015.

[21] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM Comput. Surv. 31 (1999) 264–323.

[22] R. Jensen, N.M. Parthalin, Towards scalable fuzzy-rough feature selection, Inf. Sci. 323 (2015) 1–15.

[23] H. Liu, H.J. Lu, J. Yao, Identifying relevant databases for multidatabase mining, in: Proceedings of the 2th Pacific-Asia Conference on Knowledge Discovery and Data Mining, 1998, pp. 210–221.

[24] J.N.K. Liu, Y.X. Hu, Y.L. He, A set covering based approach to find the reduct of variable precision rough set, Inf. Sci. 275 (2014a) 83–100.

[25] X. Liu, Y.H. Qian, J.Y. Liang, A rule-extraction framework under multigranulation rough sets, Int. J. Mach. Learn. Cybern. 5 (2014b) 319–326.

[26] Y. Luo, D.C. Tao, B. Geng, C. Xu, S.J. Maybank, Manifold regularized multitask learning for semi-supervised multilabel image classification, IEEE Trans. Image Process. 22 (2013) 523–536.

[27] J. Macqueen, Some methods for classification and analysis of multivariate observations, in: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, 1967, pp. 281–297.

[28] G. Madjarov, D. Gjorgjevikj, S. Dzeroski, Two stage architecture for multi-label learning, Pattern Recognit. 45 (2012) 1019–1034.

[29] R. anculef, I. Flaounas, N. Cristianini, Efficient classification of multi-labeled text streams by clashing, Expert Syst. Appl. 41 (2014) 5431–5450.

[30] N.M. Parthalin, R. Jensen, Unsupervised fuzzy-rough set-based dimensionality reduction, Inf. Sci. 229 (2013) 106–121.

[31] Y.H. Qian, Q. Wang, H.H. Cheng, J.Y. Liang, C.Y. Dang, Fuzzy-rough feature selection accelerator, Fuzzy Sets Syst. 258 (2015) 61–78.

[32] S. Ramanna, A.H. Meghdadi, J.F. Peters, Nature-inspired framework for measuring visual image resemblance: A near rough set approach, Theor. Comput. Sci. 412 (2011) 5926–5938.

[33] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, Mach. Learn. 85 (2011) 333–359.

[34] J. Read, A pruned problem transformation method for multi-label classification, in: Proceedings of 2008 New Zealand Computer Science Research Student Conference, 2008, pp. 143–150.

[35] J. Read, P. Reutemann, MEKA: a multi-label extension to WEKA. < http://meka.sourceforge.net/#download > , 2012.

[36] G.L. Ritter, H.B. Woodruff, S.R. Lowry, T.L. Isenhour, An algorithm for a selective nearest-neighbor decision rule, IEEE Trans. Inf. Theory 21 (1975) 665–669.

[37] G. Salton, Developments in automatic text retrieval, Science 253 (1991) 974–980.

[38] B.K. Sarkar, S.S. Sana, K. Chaudhuri, A genetic algorithm-based rule extraction system, Appl. Soft Comput. 12 (2012) 238–254.

[39] R.E. Schapire, Y. Singer, Improved boosting algorithms using confidence-rated predictions, Mach. Learn. 37 (1999) 297–336.

[40] R.E. Schapire, Y. Singer, Boostexter: a boosting-based system for text categorization, Mach. Learn. 39 (2000) 135–168.

[41] F. Sebastiani, Machine learning in automated text categorization, ACM Comput. Surv. 34 (2002) 1–47.

[42] W.H. Shu, H. Shen, Incremental feature selection based on rough set in dynamic incomplete data, Pattern Recognit. 47 (2014) 3890–3906.

[43] G. Tsoumakas, I. Katakis, Multi-label classification: an overview, Int. J. Data Warehous. Min. 3 (2007) 12–16.

[44] G. Tsoumakas, I. Katakis, I. Vlahavas, Mining multi-label data, in: Data Mining and Knowledge Discovery Handbook, Springer, 2010, pp. 667–685.

[45] C.Z. Wang, M.W. Shao, B.Q. Sun, Q.H. Hu, An improved attribute reduction scheme with covering based rough sets, Appl. Soft Comput. 26 (2015a) 235–243.

[46] G.Y. Wang, X.A. Ma, H. Yu, Monotonic uncertainty measures for attribute reduction in probabilistic rough set model, Int. J. Approx. Reason. 59 (2015b) 41–67.

[47] W. Wei, J.H. Wang, J.Y. Liang, X. Mi, C.Y. Dang, Compacted decision tables based attribute reduction, Knowledge-Based Syst. 86 (2015) 261–277.

[48] D.R. Wilson, T.R. Martinez, Reduction techniques for instance-based learning algorithms, Mach. Learn. 38 (2000) 257–286.

[49] B.Y. Wu, S.W. Lyu, B.G. Hu, Q. Ji, Multi-label learning with missing labels for image annotation and facial action unit recognition, Pattern Recognit. 48 (2015) 2279–2289.

[50] J.S. Wu, S.J. Huang, Z.H. Zhou, Genome-wide protein function prediction through multi-instance multi-label learning, ACM/IEEE Trans. Comput. Biol. Bioinform. 11 (2014) 891–902.

[51] X.B. Yang, Y.S. Qi, X.N. Song, J.Y. Yang, Test cost sensitive multigranulation rough set: model and minimal cost selection, Inf. Sci. 250 (2013) 184–199.

[52] X.B. Yang, Y. Qi, H.L. Yu, X.N. Song, J.Y. Yang, Updating multigranulation rough approximations with increasing of granular structures, Knowledge-Based Syst. 64 (2014) 59–69.

[53] Y. Yu, W. Pedrycz, D.Q. Miao, Neighborhood rough sets based multi-label classification for automatic image annotation, Int. J. Approx. Reason. 54 (2013) 1373–1387.

[54] Y. Yu, W. Pedrycz, D.Q. Miao, Multi-label classification by exploiting label correlations, Expert Syst. Appl. 41 (2014) 2989–3004.

[55] A.P. Zeng, T.R. Li, D. Liu, J.B. Zhang, H.M. Chen, A fuzzy rough set approach for incremental feature selection on hybrid information systems, Fuzzy Sets Syst. 258 (2015) 39–60.

[56] L.J. Zhang, Q.H. Hu, J. Duan, X.X. Wang, Multi-label feature selection with fuzzy rough sets, in: Proceedings of the 9th International Conference on Rough Sets and Knowledge Technology, Shanghai, 2014a, pp. 121–128.

[57] L.J. Zhang, Q.H. Hu, Y.C. Zhou, X.X. Wang, Multi-label attribute evaluation based on fuzzy rough sets, in: Proceedings of the 9th International Conference on Rough Sets and Current Trends in Computing, Granada and Madrid, 2014b, pp. 100–108.

[58] M.L. Zhang, Z.H. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (2014) 1819–1837.

[59] M.L. Zhang, Z.H. Zhou, ML-KNN: A lazy learning approach to multi-label learning, Pattern Recognit. 40 (2007) 2038–2048.

[60] M.L. Zhang, J.M. Pena, V. Robles, Feature selection for multi-label naive bayes classification, Inf. Sci. 179 (2009) 3218–3229.

[61] M.L. Zhang, LIFT: multi-label learning with label-specific features, in: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, 2011, pp. 1609–1614.

[62] M.L. Zhang, L. Wu, LIFT: multi-label learning with label-specific features, IEEE Trans. Pattern Anal. Mach. Intell. 37 (2015) 107–120.

[63] K. Zheng, J. Hu, Z.F. Zhan, J. Ma, J. Qi, An enhancement for heuristic attribute reduction algorithm in rough set, Expert Syst. Appl. 41 (2014) 6748–6754.