Applied Intelligence (2023) 53:3017–3033
https://doi.org/10.1007/s10489-022-03386-7

Learning label-specific features with global and local label correlation for multi-label classification

Wei Weng[1] · Bowen Wei[1] · Wen Ke[1] · Yuling Fan[2] · Jinbo Wang[3] · Yuwen Li[4]

Accepted: 12 February 2022 / Published online: 19 May 2022

Abstract

Multi-label algorithms often use an identical feature space to build classification models for all labels. However, labels generally express different semantic information and should have their own characteristics. A few algorithms have been proposed to find label-specific features to construct discriminative classification models. Some use global label correlation to make the reconstructed features more discriminative, but they usually neglect the local correlation between labels. To solve this problem, we propose a new algorithm, named learning Label-specific Features with Global and Local label Correlation (LFGLC). The algorithm integrates both global and local label correlation to extract label-specific features for each label. Specifically, global label correlation is calculated by the label co-occurrence frequency between label pairs, and local label correlation is learned from the neighborhood of each instance. Comprehensive experiments on 12 multi-label data sets clearly manifest that the proposed algorithm performs competitively in feature selection and multi-label classification.

Keywords Multi-label classification · Label-specific features · Global label correlation · Local label correlation · Feature selection

1 Introduction

Multi-label learning aims to predict multiple labels simultaneously for an instance. It has received considerable attention due to its applications in a wide range of domains, such as image recognition, natural language processing, and complex networks. For instance, multi-label learning for fundus images can effectively improve the accuracy of diagnosis [1]; with increasing information on the internet, multi-label learning for online text can enhance retrieval accuracy [2]; and multiple labels on user profiles can be helpful for individual recommendations and marketing on social networks [3].

Compared with single-label problems, multi-label data often have large feature spaces. The number of features can reach tens of thousands when describing semantics [4–6], and some can be redundant or irrelevant for classification tasks. Moreover, high-dimensional feature space often brings negative impacts on classification tasks. Therefore, a number of algorithms concentrate on feature compression techniques to obtain a low-dimension expression of multi-label data, effectively improving the performance of multi-label classification. Mutual information is widely used for feature compression, which enables efficient performance in multi-label classification [7, 8]. However,

Wei Weng and Bowen Wei contributed equally to this work.

✉ Wei Weng
xmutwei@163.com

Bowen Wei
wbwjohn@163.com

Wen Ke
wenkk7@163.com

Yuling Fan
ylingfan@126.com

Jinbo Wang
wangjb168@163.com

Yuwen Li
liyuwen@seu.edu.cn

[1] Department of Computer and Information Engineering, Xiamen University of Technology, Xiamen, 361024, China

[2] School of Automation, Xiamen University, Xiamen, 361005, China

[3] School of Economics, Xiamen University, Xiamen, 361005, China

[4] School of Instrument Science and Engineering, Southeast University, Nanjing, 210096, China

most feature compression algorithms construct an identical low-dimensional feature space for all labels [9, 10]. In other words, different labels share the same feature expression. In multi-label problems, labels reflect different semantics. Therefore, labels have unique features, named label-specific features, which can be used to distinguish them. These features are mostly related to corresponding labels and they are the most appropriate to distinguish labels. Zhang and Wu [11] introduced the concept of label-specific features in the method LIFT for the first time in 2015. Although there are some variants of LIFT, such as LETTER [12], LSDM [13], LF-LPLC [14], the research of label-specific features is ongoing.

In multi-label problems, labels do not occur independently but present some dependence. In other words, some labels tend to appear together in many instances, while others rarely co-occur. Using label correlation is conducive to learning the more efficient and robust classification model [15]. Labels sometimes have few positive instances, in which case it is important to use label correlation. To make full use of label correlation has become a major research direction in multi-label classification, and is important in many algorithms [16–19]. Considering the complexity of real-world label correlation, some label correlations are global and some others may be local. Although most existing algorithms have considered global [20–22] or local [15, 23, 24] label correlation for multi-label learning, both global and local label correlation are less taken into account.

As discussed above, label-specific features and label correlation are two important characteristics for multi-label learning. We unify these and propose label-specific features with global and local label correlation (LFGLC), which calculates global and local label correlation by label co-occurrence and neighborhood information, respectively, and adds label correlation to linear regression with the ℓ_1-norm to learn label-specific features for each label.

LFGLC makes the following contributions:

1. LFGLC integrates both global and local label correlation to select label-specific features. To the best of our knowledge, this is the first work to make reconstructed features more discriminative.
2. Linear regression modeled by LFGLC can be simultaneously applied to multi-label classification and feature selection.
3. Experiments on multiple data sets with different sizes and domains show that LFGLC outperforms several multi-label classification and feature selection algorithms in terms of both example-based evaluation metrics and label-based evaluation metrics.

The rest of this article is organized as follows. Section 2 reviews related work on label correlation and label-specific features for multi-label learning. Section 3 describes LFGLC. Experimental results and analysis are shown in Section 4. Section 5 presents conclusions.

2 Related work

Our work is related to label correlation and label-specific features for multi-label learning. We present related algorithms on label correlation and label-specific features based algorithms.

2.1 Label correlation

Making full use of label correlation is a major research direction of multi-label classification. [16–19, 25]. Label correlation strategies can be divided into three types [26]: first-order, second-order and high-order. First-order algorithms ignore label correlation completely, including BR [27], LIFT [11] and ML*k*NN [28]. Second-order algorithms consider pairwise label correlation, such as CPNL [29], PCT [30] and GBRAML [31]. High-order algorithms consider the correlation between more than two labels, such as CC [20], BNCC [32] and MLMF [33].

In terms of relation extraction or the use of perspective, label correlation based algorithms can be categorized as global, local, and global-local combined relation algorithms.

2.1.1 Global relation algorithm

A global relation algorithm, such as CC [20], CLSF [21] and A-GCN [22], assumes that label correlation is global. In other words, the correlation between labels exists in all training data. For example, CC puts all labels in a random sequence. Binary classifier outputs of previous labels are added to a label's original feature space as new features, and the corresponding binary classifiers are learned in accordance with this sequence. Label correlation is constructed on all training data. A-GCN uses a label graph to learn global label correlations with word embeddings.

2.1.2 Local relation algorithm

In a local relation algorithm, label correlation exists in part of the training data. For example, "apple" and "fruit" have a strong relation in gourmet magazines, while "apple" and "digital equipment" often occur together in technical journals. Obviously, the dependence relations of labels only exist in some data in this case. If such label correlation is extracted or used from the global perspective, unnecessary and even misguided constraints will be imposed over all instances, which will decrease the performance of classification models. LPLC [15] considers label correlation locally, finding the positive and negative label correlation of each label for all training instances. Then, for each testing instance, the maximum posterior probability is used for prediction based on the local positive and negative label correlation of its k-nearest neighbors. Ma [23] divides the training data into several groups, whose instances share different label correlations.

2.1.3 Global-local combined relation algorithm

Global-local combined relation algorithms consider both global and local label correlation to establish a high-efficiency classification model. For example, GLOCAL [34] learns both global and local label correlation by manifold regularization. GLkEL [35] selects the most correlated k-labelsets from label space by approximated joint mutual information to evaluate global label correlation. Then, it clusters the training data into different groups and evaluates the local label correlation in each group.

2.2 Label-specific features

Multi-label algorithms often use identical feature expressions to build classification models for different labels. In other words, different labels use the same feature matrix in the learning process. However, the LIFT [11] algorithm assumes that labels have unique expressions. The concept of label-specific features has evident differences from the concept of traditional feature compression.

At present, there are two main methods to construct label-specific features. One is feature extraction and the other is feature selection. The former one is represented by LIFT, while the latter one is represented by LLSF [36].

2.2.1 Feature extraction based label-specific features

LIFT [11] extracts label-specific features for each label through feature extraction. Specifically, instances related to any label are viewed as positive instances, and other instances as negative. K-means [37] is utilized to cluster on the positive and negative instance sets. Distances from each original instance to the centers of these clusters are calculated, which form the new features. Next, binary classifiers are learned on these label-specific features. For different labels, distributions of positive and negative instances are different so that the reconstructed label-specific features vary from each other. Extensive experiments have demonstrated the effect of LIFT, resulting in the proposal of a number of algorithms.

Based on LIFT, LF-LPLC [14] integrates label-specific features and local pairwise label correlation, where the specific features of each label are expanded by uniting the related features from correlated labels. This enriches the labels' semantic information and somewhat solves the class-imbalance problem. LETTER [12] extracts label-specific features from instance and feature levels. From the instance level, sparse and prototype constraints are used to find more discriminative instance centers. From the feature level, clustering is utilized to find feature centers from the original features of positive and negative instances. The final label-specific features are composed of centers extracted from the above two levels. Related work includes LSDM [13], ELIFT [38], and so on.

2.2.2 Feature selection based label-specific features

The above algorithms all adopt feature transformation to extract label-specific features. However, the LLSF algorithm proposed by Huang [36] learns label-specific features through a feature selection technique. LLSF assumes that each label is only related to some of the original features, and it expresses such sparsity in linear regression with ℓ_1 constraint. Nonzero regression parameters indicate that the corresponding features are label-specific, and other features are not.

The objective function of LLSF assumes that strongly correlated labels have more label-specific features than weakly correlated labels. Since LLSF implements feature selection through linear regression, it can learn binary classification models based on the selected features. MCUL [39] also utilizes ℓ_1-norm regularization on the coefficient matrix to learn sparse label-specific features, so as to deal with missing and completely unobserved labels. NSLSF [40] considers that the sparsity assumption does not hold in some applications, and proposes a feature selection based approach to select label-specific features. It translates logic labels to numeric labels to convey more semantic information and embeds the label correlation. Linear regression with ℓ_1 constraint describes the discrimination of label-specific features based on the numeric labels.

3 Proposed algorithm

Given a multi-label data set $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)|1 \leq i \leq n\}$ with n instances, we denote feature set $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]^T \in R^{n\times d}$, where d is the dimension of features. And let $Y = [\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n]^T \in \{0, 1\}^{n\times l}$ denotes label set, where l is the number of labels. If $y_{ij} = 1$, instance $\boldsymbol{x}_i$ belongs to label $\boldsymbol{y}_j$, and otherwise, $y_{ij} = 0$. LFGLC aims to use linear regression with the ℓ_1-norm to find label-specific features for each label, based on which it transforms multi-label classification to several binary classifications. To further improve the performance of classification, both global and local label correlation are taken into account. Label-specific features selected from the original features can achieve higher discriminability for classification.

As shown in Fig. 1, the main process of LFGLC can be summarized as the following three parts: global label correlation calculation, local label correlation calculation and label-specific feature selection. In global label correlation calculation, pairwise label correlation is calculated by the label co-occurrence frequency. In local label correlation calculation, the label correlation is calculated according to the instance and its neighbors. In label-specific feature selection, linear regression with the ℓ_1-norm on parameters and constraints on label correlations is employed to select features for each label.

3.1 Label-specific feature selection

As mentioned, label-specific features are more discriminative, so as to construct effective classification models for all labels. We use linear regression with the ℓ_1-norm to find label-specific features, as introduced in LLSF [36]. The objective function can be formulated as:

$$\min_{W} L(W) + R(W), \tag{1}$$

where $L(W)$ is formulated as:

$$L(W) = \frac{1}{2}\sum_{i=1}^{n} \|\boldsymbol{x}_i W + \boldsymbol{b} - \boldsymbol{y}_i\|_2, \tag{2}$$

where $W = [\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_l] \in R^{d\times l}$ denotes the coefficient of linear regression, and $\boldsymbol{b} = [b_1, b_2, \ldots, b_l] \in R^{1\times l}$ denotes the bias of linear regression. The bias $\boldsymbol{b}$ can be added to the coefficient W when the constant value 1 is added as an additional dimension to feature set X. Then $L(W)$ can be simplified to:

$$\begin{aligned} L(W) &= \frac{1}{2}\sum_{i=1}^{n} \|\boldsymbol{x}_i W - \boldsymbol{y}_i\|_2 \\ &= \frac{1}{2} Tr((XW - Y)^T D(XW - Y)), \end{aligned} \tag{3}$$

where D is a diagonal matrix and its diagonal element d_{ii} is formulated as:

$$d_{ii} = \frac{1}{\|\boldsymbol{x}_i W - \boldsymbol{y}_i\|_2}. \tag{4}$$

To select features for each label, the ℓ_1-norm is added to the coefficient W,

$$R(W) = \|W\|_1, \tag{5}$$

and can make the coefficient W sparse. For each $\boldsymbol{w}_i = [w_{i1}, w_{i2}, \ldots, w_{id}]^T$, the value of w_{ij} indicates the discriminability of the j-th feature to label $\boldsymbol{y}_i$. If w_{ij}=0, then the j-th feature is not helpful to label $\boldsymbol{y}_i$, and otherwise it can be regarded as a label-specific feature to label $\boldsymbol{y}_i$.

The above label-specific feature selection does not consider label correlation. We next will utilize both global and local label correlation to further constrain the coefficient.

3.2 Global label correlation calculation

As introduced in Section 2, labels do not occur independently but present some dependence in multi-label problems. Exploiting label correlation can effectively improve the performance of multi-label classifiers. Similar to LLSF [36], we assume that two strongly correlated labels may share more label-specific features than weakly correlated labels. Then the inner product between the corresponding

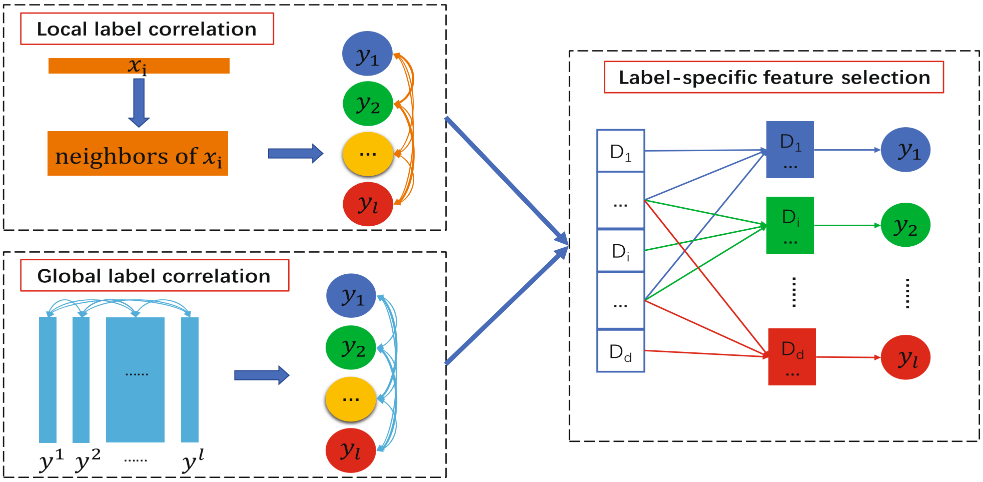


Fig. 1 Global and local label correlation, label-specific features for each label are represented in different colors. The thickness of the connection between labels indicates the strength of correlation. Note: For ease of reading and understanding of these images, please read the corresponding online version of the paper (colors have been retained)

coefficients of labels will be large when these labels are strongly correlated, and otherwise it will be small. Here we denote $GC(W)$ as global label correlation, as shown in (6).

$$GC(W) = \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} c_{ij}\boldsymbol{w}_i^T\boldsymbol{w}_j = \frac{1}{2}Tr(WCW^T), \tag{6}$$

$$c_{ij} = \frac{|\boldsymbol{y}^i \Delta \boldsymbol{y}^j|}{|\boldsymbol{y}^i \cup \boldsymbol{y}^j|}, \tag{7}$$

where c_{ij} is the correlation coefficient between label $\boldsymbol{y}_i$ and label $\boldsymbol{y}_j$. $\boldsymbol{y}^i$ and $\boldsymbol{y}^j$ are the i-th column and the j-th column of Y. It can be seen from (7), the value of c_{ij} will be small when there are more co-occurrence labels between $\boldsymbol{y}^i$ and $\boldsymbol{y}^j$, and otherwise it is large.

3.3 Local label correlation calculation

As discussed in Section 2, label correlation also exists in some of instances. A classification model that considers local label correlation can be more suitable to real-world problems. Motivated by previous work [28], instance may have more similar labels with its neighbors. The proposed algorithm finds the k-nearest neighbors of each instance by Euclidean distance firstly. Then in the neighborhood of an instance, the probabilities of labels are calculated by (8).

$$\boldsymbol{N}_i = \frac{1}{k}\sum_{i=1}^{k}\boldsymbol{y}_{N_i}, \tag{8}$$

where $\boldsymbol{N}_i$ denotes the probabilities of labels in the neighborhood of an instance, and $\boldsymbol{y}_{N_i}$ denotes the labels of the i-th neighbor. Based on (8), local label correlation $LC(W)$ can be formulated as:

$$\begin{aligned} LC(W) &= \sum_{i=1}^{n}\|\boldsymbol{x}_i W - \boldsymbol{N}_i\|_2^2 \\ &= \|XW - N\|_F^2. \end{aligned} \tag{9}$$

3.4 Optimization via accelerated proximal gradient

According to the definition of each term, we unify label-specific feature selection and label correlation, and can rewrite the whole objective function of LFGLC as:

$$\begin{aligned} F(W) = &\frac{1}{2}Tr((XW-Y)^T D(XW-Y)) \\ &+ \frac{\alpha}{2}Tr(WCW^T) + \frac{\beta}{2}\|XW - N\|_F^2 + \gamma\|W\|_1, \end{aligned} \tag{10}$$

where α, β and γ are nonnegative parameters that control the contribution of each term. The objective function is a convex optimization problem. To solve the nonsmoothness caused by ℓ_1-norm, the accelerated proximal gradient method is employed to optimize this objective function. General accelerated proximal gradient method can divide the objective function into the following two parts [43]:

$${}^{min}_{\;W} F(W) = f(W) + g(W), \tag{11}$$

$f(W)$ and $g(W)$ are convex, but $f(W)$ is smooth while $g(W)$ is nonsmooth. $f(W)$ holds Lipschitz continuous gradient: $\|\nabla f(W_1) - \nabla f(W_2)\| \leq L_f\|W_1 - W_2\|$, where L_f is the Lipschitz constant. $f(W)$ and $g(W)$ are formulated as:

$$\begin{aligned} f(W) = &\frac{1}{2}Tr((XW-Y)^T D(XW-Y)) \\ &+ \frac{\alpha}{2}Tr(WCW^T) + \frac{\beta}{2}\|XW - N\|_F^2, \end{aligned} \tag{12}$$

$$g(W) = \gamma\|W\|_1. \tag{13}$$

$\nabla f(W)$ denotes the derivative of $f(W)$ and it can be calculated by:

$$\nabla f(W) = X^T DXW - X^T DY + \alpha WC + \beta(X^T XW - X^T N). \tag{14}$$

Given W_1 and W_2, we have

$$\begin{aligned} &\|\nabla f(W_1) - \nabla f(W_2)\|_F^2 \\ &= \|X^T DX\Delta W + \alpha\Delta WC + \beta X^T X\Delta W\|_F^2 \\ &\leq 3\|X^T DX\Delta W\|_F^2 + 3\|\alpha\Delta WC\|_F^2 + 3\|\beta X^T X\Delta W\|_F^2 \\ &\leq 3\|X^T DX\|_2^2\|\Delta W\|_F^2 + 3\|\alpha C\|_2^2\|\Delta W\|_F^2 \\ &+ 3\|\beta X^T X\|_2^2\|\Delta W\|_F^2 \\ &= 3(\|X^T DX\|_2^2 + \|\alpha C\|_2^2 + \|\beta X^T X\|_2^2)\|W\|_F^2, \end{aligned} \tag{15}$$

thus, the Lipschitz constant L_f can be calculated as:

$$L_f = \sqrt{3(\|X^T DX\|_2^2 + \|\alpha C\|_2^2 + \|\beta X^T X\|_2^2)}. \tag{16}$$

The pseudocode of the optimization of LFGLC is summarized in Algorithm 1. Steps 6-12 are the iteration process of the accelerated proximal gradient method. Previous work [42] showed that for a sequence b_t satisfying $b_t^2 - b_t \leq b_{t-1}^2$, the convergence rate can be improved to $\mathcal{O}(t^{-2})$ when letting $W^{(t)} = W_t + \frac{b_{(t-1)}-1}{b_t}(W_t - W_{t-1})$, where W_t is the coefficient W obtained at the t-th iteration. The ℓ_1-norm can be solved by the soft-thresholding operator $S_\epsilon[w]$ in each iteration, and the step size ϵ of the soft-thresholding operator is set to $\frac{\gamma}{L_f}$ in this optimization process.

After learning the coefficient W, the prediction of the test data X_t can be calculated by $sign(S_t - \tau)$. $S_t = X_t W$, τ is the threshold which is set to 0.5 in LFGLC. The pseudocode of the test procedure is summarized in Algorithm 2.

Algorithm 1 Optimization of LFGLC.

```
Input:
$X$: $n \times d$ Training data matrix.
$Y$: $n \times l$ Label matrix associated with $X$.
$\alpha, \beta, \gamma, \eta$: Weighting parameters.
Output:
$W$: $d \times l$ Coefficient matrix.
1: Initialization:
2:     $b_0, b_1 \leftarrow 1; t \leftarrow 1; W_0, W_1 \leftarrow (X^T X + \eta I)^{-1} X^T Y$;
3: Compute correlation coefficient $C$ according to (7);
4: Compute N according to (9);
5: repeat
6:     Compute the diagonal matrix $D$ according to (4);
7:     Compute $L_f$ according to (16);
8:     $W^{(t)} \leftarrow W_t + \frac{b_{(t-1)}-1}{b_t}(W_t - W_{t-1})$;
9:     $G^{(t)} \leftarrow W^{(t)} - \frac{1}{L_f}\nabla f(W^{(t)})$;
10:    $W_{t+1} \leftarrow S_{\frac{\gamma}{L_f}}(G^{(t)})$;
11:    $b_{t+1} \leftarrow \frac{1+\sqrt{4b_t^2+1}}{2}$;
12:    $t \leftarrow t + 1$;
13: until stop criterion reached
14: $W \leftarrow W_{t+1}$;
```

Algorithm 2 Test of LFGLC.

```
Input:
$X_t$: $n_t \times d$ Testing data matrix.
$W$: $d \times l$ Coefficient matrix.
$\tau$: Threshold.
Output:
$Y_t$: $n_t \times l$ Predicted label matrix.
$S_t$: $n_t \times l$ Score matrix.
1: $S_t \leftarrow X_t W$;
2: $Y_t \leftarrow sign(S_t - \tau)$;
```

3.5 Discussion

Note that LFGLC is similar to LLSF [36] and JFSC [41]. LLSF uses cosine similarity to calculate global label correlation and the correlation among labels is added to linear regression with the ℓ_1-norm to select label-specific features for each label. Based on LLSF, JFSC adds a Fisher discriminant-based regularization term to obtain a large inter-class distance and a small inner-class distance for classification. These algorithms perform competitively on label-specific feature selection and classification, but they neglect local label correlation. Different from them, LFGLC is devoted to take both global and local label correlation into consideration. Global label correlation is calculated by the label co-occurrence frequency between label pairs, and local label correlation is calculated by each instance with its neighbors. Both are added to linear regression with the ℓ_1-norm to select more discriminative label-specific features for multi-label classification.

Table 1 Experimental data sets

Data set	#Instance	#Feature	#Label	Cardinality	Domain
flags	194	19	7	3.392	images
cal500	502	68	174	26.044	music
emotions	593	72	6	1.869	music
genbase	662	1185	27	1.252	biology
medical	978	1449	45	1.245	text
image	2000	294	5	1.236	images
yeast	2417	103	14	4.237	biology
health	5000	612	32	1.662	text
corel5k	5000	499	374	3.522	images
arts	5000	462	26	1.636	text(web)
education	5000	550	33	1.461	text(web)
corel16k001	13766	500	153	2.859	images

3.6 Complexity analysis

The complexity of the optimization of LFGLC includes initialization and iteration. In initialization, the step of initializing coefficient W_1 has complexity $\mathcal{O}(nd^2 + d^3 + ndl + d^2l)$. Step 2 calculates the global label correlation matrix C, and it needs $\mathcal{O}(nl^2)$. In step 3, the calculation of N includes finding the k-nearest neighbors of each instance and calculating the probabilities of labels in the neighborhood, which needs $\mathcal{O}(n^2d + nkd)$. In iteration, step 5 calculates the diagonal matrix D, with complexity $\mathcal{O}(ndl)$. The calculation of the Lipschitz constant L_f in step 6 has complexity $\mathcal{O}(nd^2 + d^3 + l^3)$. To calculate the derivative of $f(W)$ in step 8 has complexity $\mathcal{O}(nd^2 + d^2l + ndl + dl^2)$.

4 Experiments

4.1 Data sets

Experiments were conducted on 12 real-world multi-label data sets, as described in Table 1. "Cardinality" is the average number of labels of each instance in one data set. All these data sets can be obtained from mulan[1], lamda[2], and meka[3].

[1] http://mulan.sourceforge.net/datasets.html

[2] http://lamda.nju.edu.cn

[3] http://meka.sourceforge.net

4.2 Evaluation metrics

For a given testing data set $D_t = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) | 1 \leq i \leq n_t\}$, the ground-truth label of the i-th instance $\boldsymbol{x}_i$ is represented as $\boldsymbol{y}_i \in \{0, 1\}^l$, and let $\hat{\boldsymbol{y}}_i \in \{0, 1\}^l$ denotes the predicted label of the i-th instance $\boldsymbol{x}_i$. There are 7 multi-label evaluation metrics used for the evaluation of LFGLC. These evaluation metrics can be divided into the following two types: example-based evaluation metrics and label-based evaluation metrics.

Example-based evaluation metrics:

Accuracy measures Jaccard similarity between ground-truth and predicted labels:

$$Accuracy = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{|\boldsymbol{y}_i \cap \hat{\boldsymbol{y}}_i|}{|\boldsymbol{y}_i \cup \hat{\boldsymbol{y}}_i|}. \tag{17}$$

Precision is the proportion of positive labels that are predicted correctly:

$$Precision = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{|\boldsymbol{y}_i \cap \hat{\boldsymbol{y}}_i|}{|\hat{\boldsymbol{y}}_i|}. \tag{18}$$

Recall is the proportion of ground-truth positive labels that are correctly predicted:

$$Recall = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{|\boldsymbol{y}_i \cap \hat{\boldsymbol{y}}_i|}{|\boldsymbol{y}_i|}. \tag{19}$$

F_1 evaluates the harmonic mean between *Precision* and *Recall*:

$$F_1 = \frac{1}{n_t} \sum_{i=1}^{n_t} \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \tag{20}$$

Exact-Match evaluates how many times the predicted and ground-truth labels are exactly matched:

$$Exact - Match = \frac{1}{n_t} \sum_{i=1}^{n_t} |\boldsymbol{y}_i = \hat{\boldsymbol{y}}_i| \tag{21}$$

Label-based evaluation metrics: There are two metrics of averaging across the labels:

$$Macro\ F_1 = \frac{1}{l} \sum_{j=1}^{l} F_1(TP_j, FP_j, TN_j, FN_j), \tag{22}$$

$$Micro\ F_1 = \frac{1}{l} F_1 \left(\sum_{j=1}^{l} TP_j, \sum_{j=1}^{l} FP_j, \sum_{j=1}^{l} TN_j, \sum_{j=1}^{l} FN_j \right), \tag{23}$$

where TP_j, FP_j, TN_j, FN_j are the number of true positive, false positive, true negative, and false negative instances with respect to label $\boldsymbol{y}_j$ respectively.

4.3 Comparison methods

We compared the multi-label classification performance of LFGLC with several state-of-the-art algorithms:

BR [27] transforms multi-label classification to several binary classification tasks without considering label correlation, where each binary classifier corresponds to one label.

CC [20] puts all labels in a random sequence, and in accordance with each label, learns the corresponding binary classifier. For each label, the binary classifier outputs of its previous labels are added as new features.

ML*k*NN[4] [28] finds the k-nearest neighbors for each instance in Euclidean space. The maximum posterior probability of each label is used to estimate the probability based on the number of neighbors belonging to each label. The parameter k is set to 10.

Lasso [43] uses linear regression with the ℓ_1-norm to select features from the original feature space according to nonzero regression coefficients, while neglecting label correlation. Parameter α is searched in $\{2^{-10}, 2^{-9}, \ldots, 2^{10}\}$.

LLSF[5] [36] uses cosine similarity to calculate pairwise label correlation, which is added to linear regression with the ℓ_1-norm to select label-specific features for each label. Parameters α and β are searched in $\{2^{-10}, 2^{-9}, \ldots, 2^{10}\}$. ρ is searched in $\{0.1, 1, 10\}$.

Based on LLSF, JFSC [41] uses a Fisher discriminant-based regularization term to achieve a large inter-class distance and small inner-class distance for classification. Parameters α, β and γ are searched in $\{4^{-5}, 4^{-4}, \ldots, 4^5\}$. η is searched in $\{0.1, 1, 10\}$.

Based on LLSF, NSLSF [40] translates logic labels to numeric labels to convey more semantic information and embed label correlations. Parameters α and β are same as LLSF, ρ is set to 0.5.

The searching scales of parameters α, β, γ and η in LFGLC are same as JSFC. The number of k-nearest neighbors is set to 10.

BR, ML*k*NN, and Lasso are first-order algorithms that do not consider label correlation; LLSF, JFSC, and NSLSF are second-order algorithms with global label correlation; and CC can be regarded as a high-order algorithm with global label correlation. Specifically, Lasso, LLSF, JFSC, and NSLSF are feature selection based label-specific features algorithms. For fair comparisons, the parameters of these algorithms are set according to the suggestions in their original papers. LIBSVM [44] with a linear kernel is employed as the base binary classifier for BR and CC, and the parameter C is set to 1. For the sake of fairness, the threshold of LFGLC is set to 0.5, the same as other

[4] source code: http://cse.seu.edu.cn/PersonalPage/zhangml/index.html

[5] source code: http://www.escience.cn/people/huangjun/index.html

Table 2 Experimental results (mean± std(rank)) of different comparison algorithms on *Accuracy*, *Precision*, *Recall*, F_1 and *Exact-Match*

Data set	BR	CC	ML*k*NN	Lasso	LLSF	JFSC	NSLSF	LFGLC
Accuracy↑								
flags	0.506 ± 0.034(5)	0.500 ± 0.018(8)	0.505 ± 0.024(6)	0.505 ± 0.028(7)	0.511 ± 0.047(4)	0.516 ± 0.038(3)	0.521 ± 0.020(2)	**0.550 ± 0.035(1)**
cal500	0.194 ± 0.005(8)	0.203 ± 0.017(6)	0.196 ± 0.007(7)	0.215 ± 0.005(5)	0.315 ± 0.013(4)	0.316 ± 0.008(3)	0.316 ± 0.005(2)	**0.328 ± 0.007(1)**
emotions	0.480 ± 0.021(8)	0.542 ± 0.046(5)	0.544 ± 0.020(4)	0.505 ± 0.028(7)	0.521 ± 0.023(6)	0.552 ± 0.027(2)	**0.555 ± 0.023(1)**	0.545 ± 0.044(3)
genbase	0.986 ± 0.005(7)	0.987 ± 0.007(5)	0.944 ± 0.010(8)	0.987 ± 0.008(6)	0.990 ± 0.005(3)	0.990 ± 0.004(2)	0.988 ± 0.005(4)	**0.992 ± 0.004(1)**
medical	0.725 ± 0.014(7)	0.739 ± 0.017(5)	0.559 ± 0.035(8)	0.732 ± 0.022(6)	0.751 ± 0.024(3)	0.753 ± 0.015(2)	0.751 ± 0.036(4)	**0.755 ± 0.024(1)**
image	0.434 ± 0.019(7)	**0.560 ± 0.015(1)**	0.481 ± 0.016(5)	0.416 ± 0.009(8)	0.502 ± 0.018(4)	0.459 ± 0.030(6)	0.508 ± 0.017(3)	0.529 ± 0.020(2)
yeast	0.493 ± 0.008(8)	0.510 ± 0.019(3)	0.506 ± 0.009(4)	0.497 ± 0.005(7)	0.501 ± 0.008(6)	0.511 ± 0.010(2)	0.504 ± 0.004(5)	**0.513 ± 0.012(1)**
health	0.571 ± 0.013(4)	**0.610 ± 0.016(1)**	0.260 ± 0.019(8)	0.503 ± 0.010(7)	0.561 ± 0.008(6)	0.572 ± 0.006(3)	0.565 ± 0.007(5)	0.580 ± 0.017(2)
core15k	0.016 ± 0.003(7)	0.067 ± 0.002(5)	0.012 ± 0.002(8)	0.052 ± 0.004(6)	0.140 ± 0.007(4)	0.142 ± 0.005(2)	0.142 ± 0.009(3)	**0.163 ± 0.004(1)**
arts	0.263 ± 0.009(7)	0.355 ± 0.026(5)	0.091 ± 0.003(8)	0.283 ± 0.010(6)	0.373 ± 0.012(4)	0.380 ± 0.010(2)	0.376 ± 0.007(3)	**0.381 ± 0.007(1)**
education	0.260 ± 0.010(7)	0.377 ± 0.019(4)	0.230 ± 0.016(8)	0.295 ± 0.018(6)	0.365 ± 0.007(5)	0.389 ± 0.008(2)	0.377 ± 0.013(3)	**0.393 ± 0.012(1)**
corel16k001	0.011 ± 0.003(7)	0.083 ± 0.017(5)	0.004 ± 0.000(8)	0.038 ± 0.002(6)	0.149 ± 0.005(3)	0.133 ± 0.005(4)	0.149 ± 0.002(2)	**0.151 ± 0.003(1)**
Ave.rank	6.8	4.4	6.8	6.4	4.3	2.7	3.0	**1.3**
Precision↑								
flags	0.714 ± 0.041(3)	0.652 ± 0.024(8)	0.678 ± 0.012(7)	0.709 ± 0.041(5)	0.710 ± 0.044(4)	0.706 ± 0.059(6)	**0.717 ± 0.035(1)**	0.715 ± 0.047(2)
cal500	**0.632 ± 0.012(1)**	0.600 ± 0.011(3)	0.605 ± 0.013(2)	0.556 ± 0.023(4)	0.421 ± 0.010(7)	0.412 ± 0.018(8)	0.422 ± 0.010(6)	0.424 ± 0.011(5)
emotions	0.621 ± 0.026(7)	0.654 ± 0.050(5)	**0.706 ± 0.022(1)**	0.649 ± 0.031(6)	0.619 ± 0.031(8)	0.675 ± 0.017(2)	0.655 ± 0.026(4)	0.657 ± 0.035(3)
genbase	0.994 ± 0.005(5)	0.995 ± 0.005(3)	0.975 ± 0.010(8)	0.991 ± 0.005(6)	0.994 ± 0.003(4)	**0.996 ± 0.004(1)**	0.991 ± 0.008(7)	0.995 ± 0.004(2)
medical	0.777 ± 0.010(6)	0.786 ± 0.013(2)	0.614 ± 0.030(8)	0.770 ± 0.017(7)	0.778 ± 0.024(5)	**0.789 ± 0.008(1)**	0.786 ± 0.031(3)	0.785 ± 0.022(4)
image	0.491 ± 0.015(7)	**0.634 ± 0.015(1)**	0.549 ± 0.011(5)	0.471 ± 0.012(8)	0.563 ± 0.016(3)	0.515 ± 0.034(6)	0.563 ± 0.021(4)	0.585 ± 0.012(2)
yeast	0.718 ± 0.016(2)	0.673 ± 0.018(6)	**0.720 ± 0.013(1)**	0.711 ± 0.013(3)	0.666 ± 0.007(8)	0.688 ± 0.010(5)	0.669 ± 0.008(7)	0.692 ± 0.008(4)
health	0.671 ± 0.010(2)	**0.714 ± 0.022(1)**	0.325 ± 0.020(8)	0.595 ± 0.009(7)	0.644 ± 0.011(6)	0.652 ± 0.005(4)	0.650 ± 0.005(5)	0.665 ± 0.016(3)
core15k	0.045 ± 0.008(7)	0.175 ± 0.012(5)	0.026 ± 0.003(8)	0.137 ± 0.012(6)	0.260 ± 0.006(3)	**0.265 ± 0.006(1)**	0.264 ± 0.014(2)	0.237 ± 0.004(4)
arts	0.322 ± 0.014(7)	0.426 ± 0.037(5)	0.115 ± 0.004(8)	0.343 ± 0.010(6)	0.440 ± 0.013(4)	0.448 ± 0.009(2)	0.442 ± 0.008(3)	**0.450 ± 0.008(1)**
education	0.306 ± 0.010(7)	0.444 ± 0.023(3)	0.271 ± 0.020(8)	0.347 ± 0.022(6)	0.421 ± 0.006(5)	0.445 ± 0.012(2)	0.429 ± 0.019(4)	**0.454 ± 0.013(1)**
corel16k001	0.027 ± 0.006(7)	0.199 ± 0.036(5)	0.009 ± 0.001(8)	0.091 ± 0.002(6)	0.257 ± 0.005(2)	0.235 ± 0.006(4)	**0.257 ± 0.004(1)**	0.239 ± 0.004(3)
Ave.rank	5.0	4.0	6.0	5.8	4.9	3.5	3.9	**2.8**

Table 2 (continued)

Data set	BR	CC	ML*k*NN	Lasso	LLSF	JFSC	NSLSF	LFGLC
Recall↑								
flags	0.618 ± 0.043(7)	0.638 ± 0.030(3)	0.648 ± 0.048(2)	0.616 ± 0.031(8)	0.620 ± 0.045(6)	0.632 ± 0.023(4)	0.629 ± 0.037(5)	**0.677 ± 0.028(1)**
cal500	0.218 ± 0.006(8)	0.234 ± 0.024(6)	0.224 ± 0.009(7)	0.263 ± 0.007(5)	0.554 ± 0.028(4)	0.575 ± 0.014(2)	0.555 ± 0.017(3)	**0.589 ± 0.006(1)**
emotions	0.552 ± 0.030(8)	0.647 ± 0.049(5)	0.616 ± 0.029(6)	0.593 ± 0.033(7)	0.654 ± 0.030(2)	0.653 ± 0.034(3)	**0.681 ± 0.032(1)**	0.653 ± 0.050(4)
genbase	0.988 ± 0.004(7)	0.989 ± 0.007(6)	0.945 ± 0.011(8)	0.991 ± 0.007(5)	0.993 ± 0.003(2)	0.992 ± 0.004(3)	0.991 ± 0.005(4)	**0.995 ± 0.002(1)**
medical	0.759 ± 0.021(7)	0.766 ± 0.021(6)	0.576 ± 0.036(8)	0.783 ± 0.022(5)	**0.836 ± 0.014(1)**	0.825 ± 0.009(3)	0.809 ± 0.036(4)	0.827 ± 0.014(2)
image	0.458 ± 0.019(7)	0.580 ± 0.018(2)	0.496 ± 0.024(5)	0.441 ± 0.015(8)	0.565 ± 0.025(4)	0.491 ± 0.025(6)	0.566 ± 0.020(3)	**0.604 ± 0.024(1)**
yeast	0.565 ± 0.007(8)	0.607 ± 0.022(5)	0.577 ± 0.009(6)	0.576 ± 0.004(7)	0.618 ± 0.015(2)	0.613 ± 0.007(3)	**0.621 ± 0.008(1)**	0.608 ± 0.015(4)
health	0.593 ± 0.013(6)	0.622 ± 0.016(5)	0.281 ± 0.021(8)	0.529 ± 0.011(7)	0.640 ± 0.007(4)	0.657 ± 0.012(2)	0.643 ± 0.012(3)	**0.670 ± 0.015(1)**
core15k	0.016 ± 0.003(7)	0.070 ± 0.002(5)	0.013 ± 0.002(8)	0.056 ± 0.004(6)	0.195 ± 0.017(3)	0.190 ± 0.009(4)	0.199 ± 0.014(2)	**0.292 ± 0.005(1)**
arts	0.266 ± 0.009(7)	0.361 ± 0.026(5)	0.092 ± 0.003(8)	0.289 ± 0.011(6)	0.425 ± 0.013(2)	0.425 ± 0.018(3)	**0.448 ± 0.013(1)**	0.422 ± 0.006(4)
education	0.264 ± 0.009(7)	0.380 ± 0.018(5)	0.232 ± 0.016(8)	0.309 ± 0.019(6)	0.432 ± 0.011(4)	0.463 ± 0.008(2)	**0.514 ± 0.033(1)**	0.459 ± 0.008(3)
corel16k001	0.012 ± 0.003(7)	0.087 ± 0.019(5)	0.004 ± 0.000(8)	0.039 ± 0.002(6)	0.213 ± 0.008(3)	0.187 ± 0.008(4)	0.213 ± 0.004(2)	**0.247 ± 0.007(1)**
Ave.rank	7.1	4.8	6.8	6.3	3.0	3.2	2.5	**2.0**
F_1↑								
flags	0.639 ± 0.031(5)	0.629 ± 0.015(8)	0.638 ± 0.023(7)	0.638 ± 0.022(6)	0.641 ± 0.040(4)	0.645 ± 0.036(3)	0.649 ± 0.018(2)	**0.678 ± 0.032(1)**
cal500	0.320 ± 0.006(8)	0.331 ± 0.024(6)	0.322 ± 0.010(7)	0.346 ± 0.007(5)	0.471 ± 0.014(4)	0.474 ± 0.009(2)	0.472 ± 0.005(3)	**0.486 ± 0.008(1)**
emotions	0.554 ± 0.024(8)	0.622 ± 0.045(5)	0.627 ± 0.022(3)	0.588 ± 0.027(7)	0.606 ± 0.023(6)	0.631 ± 0.026(2)	**0.638 ± 0.025(1)**	0.625 ± 0.042(4)
genbase	0.989 ± 0.003(7)	0.990 ± 0.006(5)	0.954 ± 0.011(8)	0.990 ± 0.007(6)	0.992 ± 0.003(3)	0.993 ± 0.003(2)	0.990 ± 0.005(4)	**0.994 ± 0.003(1)**
medical	0.754 ± 0.014(7)	0.765 ± 0.017(5)	0.583 ± 0.033(8)	0.762 ± 0.020(6)	0.789 ± 0.020(3)	0.790 ± 0.009(2)	0.783 ± 0.034(4)	**0.791 ± 0.019(1)**
image	0.461 ± 0.016(7)	**0.591 ± 0.016(1)**	0.509 ± 0.016(5)	0.443 ± 0.012(8)	0.544 ± 0.019(4)	0.489 ± 0.030(6)	0.546 ± 0.019(3)	0.574 ± 0.018(2)
yeast	0.604 ± 0.008(8)	0.614 ± 0.016(4)	0.612 ± 0.006(5)	0.608 ± 0.005(7)	0.612 ± 0.009(6)	0.620 ± 0.008(2)	0.615 ± 0.003(3)	**0.621 ± 0.011(1)**
health	0.610 ± 0.012(6)	**0.646 ± 0.017(1)**	0.288 ± 0.020(8)	0.541 ± 0.009(7)	0.615 ± 0.008(5)	0.627 ± 0.007(3)	0.620 ± 0.008(4)	0.639 ± 0.016(2)
core15k	0.023 ± 0.005(7)	0.096 ± 0.004(5)	0.016 ± 0.002(8)	0.075 ± 0.006(6)	0.201 ± 0.011(4)	0.202 ± 0.007(3)	0.205 ± 0.013(2)	**0.241 ± 0.004(1)**
arts	0.282 ± 0.010(7)	0.378 ± 0.029(5)	0.099 ± 0.003(8)	0.303 ± 0.010(6)	0.412 ± 0.012(4)	0.417 ± 0.012(3)	**0.421 ± 0.007(1)**	0.417 ± 0.007(2)
education	0.276 ± 0.010(7)	0.398 ± 0.019(5)	0.244 ± 0.017(8)	0.316 ± 0.019(6)	0.405 ± 0.008(4)	0.432 ± 0.008(3)	**0.437 ± 0.004(1)**	0.435 ± 0.011(2)
corel16k001	0.015 ± 0.004(7)	0.115 ± 0.023(5)	0.006 ± 0.000(8)	0.052 ± 0.002(6)	0.210 ± 0.006(3)	0.186 ± 0.006(4)	0.210 ± 0.003(2)	**0.216 ± 0.004(1)**
Ave.rank	6.4	4.5	6.9	6.3	4.1	2.9	2.5	**1.5**

Table 2 (continued)

Data set	BR	CC	ML*k*NN	Lasso	LLSF	JFSC	NSLSF	LFGLC
Exact-Match↑								
flags	0.066 ± 0.041(8)	0.087 ± 0.038(6)	0.098 ± 0.020(3)	0.077 ± 0.022(7)	0.087 ± 0.034(5)	0.098 ± 0.019(2)	0.092 ± 0.034(4)	**0.102 ± 0.045(1)**
cal500	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)	0.000 ± 0.000(4)
emotions	0.252 ± 0.010(8)	0.303 ± 0.025(2)	0.295 ± 0.019(5)	0.258 ± 0.033(7)	0.258 ± 0.024(6)	0.301 ± 0.045(3)	0.296 ± 0.021(4)	**0.304 ± 0.056(1)**
genbase	0.974 ± 0.010(7)	0.977 ± 0.010(5)	0.910 ± 0.015(8)	0.977 ± 0.016(6)	0.983 ± 0.012(2)	0.978 ± 0.011(3)	0.977 ± 0.009(4)	**0.984 ± 0.006(1)**
medical	0.641 ± 0.018(6)	0.645 ± 0.025(3)	0.488 ± 0.043(8)	0.643 ± 0.028(5)	0.638 ± 0.038(7)	0.644 ± 0.039(4)	**0.655 ± 0.044(1)**	0.650 ± 0.045(2)
image	0.355 ± 0.027(7)	**0.467 ± 0.016(1)**	0.399 ± 0.029(3)	0.337 ± 0.004(8)	0.380 ± 0.018(5)	0.371 ± 0.032(6)	0.395 ± 0.015(4)	0.399 ± 0.023(2)
yeast	0.144 ± 0.017(8)	**0.193 ± 0.018(1)**	0.170 ± 0.019(2)	0.144 ± 0.010(7)	0.147 ± 0.011(5)	0.163 ± 0.017(4)	0.145 ± 0.007(6)	0.169 ± 0.025(3)
health	0.461 ± 0.016(2)	**0.510 ± 0.016(1)**	0.183 ± 0.016(8)	0.396 ± 0.013(7)	0.406 ± 0.010(5)	0.411 ± 0.008(4)	0.404 ± 0.010(6)	0.412 ± 0.017(3)
core15k	0.002 ± 0.000(7)	0.006 ± 0.002(5)	0.001 ± 0.000(8)	0.005 ± 0.001(6)	0.009 ± 0.003(3)	**0.011 ± 0.003(1)**	0.009 ± 0.002(2)	0.007 ± 0.003(4)
arts	0.212 ± 0.007(7)	**0.296 ± 0.021(1)**	0.070 ± 0.003(8)	0.227 ± 0.008(6)	0.265 ± 0.010(4)	0.279 ± 0.005(3)	0.254 ± 0.018(5)	0.283 ± 0.007(2)
education	0.217 ± 0.011(6)	**0.319 ± 0.018(1)**	0.194 ± 0.013(8)	0.239 ± 0.015(5)	0.248 ± 0.007(4)	0.265 ± 0.011(3)	0.213 ± 0.040(7)	0.276 ± 0.014(2)
corel16k001	0.002 ± 0.001(7)	0.013 ± 0.004(5)	0.000 ± 0.000(8)	0.006 ± 0.001(6)	0.015 ± 0.002(3)	**0.015 ± 0.000(1)**	0.015 ± 0.001(2)	0.013 ± 0.001(4)
Ave.rank	6.4	2.8	6.0	6.1	4.4	3.1	4.0	**2.4**

Table 3 Experimental results (mean± std(rank)) of different comparison algorithms on *Macro* F_1 and *Micro* F_1

Data set	BR	CC	ML*k*NN	Lasso	LLSF	JFSC	NSLSF	LFGLC
Macro F_1↑								
flags	0.419 ± 0.042(7)	0.414 ± 0.032(8)	0.446 ± 0.028(6)	0.482 ± 0.047(4)	0.531 ± 0.071(2)	0.481 ± 0.005(5)	0.499 ± 0.026(3)	**0.548 ± 0.018(1)**
cal500	0.046 ± 0.001(8)	0.058 ± 0.010(6)	0.056 ± 0.003(7)	0.088 ± 0.004(5)	0.133 ± 0.008(4)	0.133 ± 0.002(2)	0.133 ± 0.004(3)	**0.157 ± 0.003(1)**
emotions	0.581 ± 0.014(8)	0.622 ± 0.038(6)	0.633 ± 0.027(5)	0.615 ± 0.028(7)	0.642 ± 0.035(4)	**0.665 ± 0.017(1)**	0.661 ± 0.036(2)	0.660 ± 0.018(3)
genbase	0.717 ± 0.034(5)	0.697 ± 0.061(7)	0.528 ± 0.024(8)	0.751 ± 0.040(3)	0.769 ± 0.025(2)	0.714 ± 0.033(6)	0.730 ± 0.042(4)	**0.769 ± 0.012(1)**
medical	0.344 ± 0.016(6)	0.335 ± 0.019(7)	0.203 ± 0.015(8)	0.363 ± 0.026(4)	0.369 ± 0.016(2)	0.362 ± 0.022(5)	0.363 ± 0.025(3)	**0.372 ± 0.016(1)**
image	0.541 ± 0.016(7)	0.591 ± 0.016(4)	0.568 ± 0.015(5)	0.527 ± 0.012(8)	0.592 ± 0.019(3)	0.551 ± 0.031(6)	0.596 ± 0.016(2)	**0.612 ± 0.019(1)**
yeast	0.319 ± 0.004(8)	0.335 ± 0.019(7)	0.361 ± 0.011(3)	0.338 ± 0.003(6)	0.381 ± 0.006(2)	0.354 ± 0.006(5)	**0.383 ± 0.006(1)**	0.357 ± 0.010(4)
health	0.280 ± 0.016(5)	0.275 ± 0.011(6)	0.160 ± 0.004(8)	0.240 ± 0.009(7)	0.282 ± 0.006(4)	0.291 ± 0.003(2)	0.288 ± 0.013(3)	**0.291 ± 0.001(1)**
core15k	0.010 ± 0.001(7)	0.018 ± 0.001(5)	0.008 ± 0.001(8)	0.016 ± 0.001(6)	0.038 ± 0.003(3)	0.038 ± 0.002(2)	0.037 ± 0.000(4)	**0.046 ± 0.002(1)**
arts	0.175 ± 0.010(6)	0.201 ± 0.015(5)	0.092 ± 0.007(8)	0.170 ± 0.005(7)	**0.239 ± 0.011(1)**	0.238 ± 0.007(2)	0.234 ± 0.003(3)	0.233 ± 0.008(4)
education	0.150 ± 0.012(6)	0.173 ± 0.015(3)	0.126 ± 0.017(7)	0.120 ± 0.004(8)	0.171 ± 0.014(4)	0.175 ± 0.013(2)	0.170 ± 0.014(5)	**0.179 ± 0.010(1)**
corel16k001	0.010 ± 0.001(7)	0.024 ± 0.000(5)	0.009 ± 0.001(8)	0.015 ± 0.001(6)	0.068 ± 0.003(3)	0.067 ± 0.002(4)	0.068 ± 0.001(2)	**0.077 ± 0.002(1)**
Ave.rank	6.6	5.7	7.4	5.9	2.8	3.5	2.9	**1.6**
Micro F_1↑								
flags	0.652 ± 0.033(7)	0.644 ± 0.019(8)	0.652 ± 0.025(6)	0.659 ± 0.023(4)	0.662 ± 0.047(3)	0.659 ± 0.032(5)	0.663 ± 0.016(2)	**0.692 ± 0.027(1)**
cal500	0.316 ± 0.005(8)	0.330 ± 0.026(6)	0.319 ± 0.011(7)	0.348 ± 0.006(5)	0.475 ± 0.015(4)	0.478 ± 0.009(2)	0.477 ± 0.005(3)	**0.491 ± 0.008(1)**
emotions	0.626 ± 0.018(8)	0.654 ± 0.036(6)	0.670 ± 0.020(4)	0.642 ± 0.022(7)	0.660 ± 0.026(5)	**0.689 ± 0.018(1)**	0.680 ± 0.029(2)	0.673 ± 0.030(3)
genbase	0.989 ± 0.005(6)	0.990 ± 0.005(5)	0.944 ± 0.002(8)	0.989 ± 0.007(7)	0.991 ± 0.005(3)	0.991 ± 0.004(2)	0.990 ± 0.004(4)	**0.994 ± 0.002(1)**
medical	0.797 ± 0.012(6)	0.794 ± 0.015(7)	0.660 ± 0.031(8)	0.804 ± 0.017(4)	0.800 ± 0.023(5)	0.806 ± 0.013(3)	**0.808 ± 0.030(1)**	0.806 ± 0.010(2)
image	0.544 ± 0.018(7)	0.588 ± 0.014(4)	0.568 ± 0.012(5)	0.530 ± 0.010(8)	0.590 ± 0.019(3)	0.555 ± 0.028(6)	0.594 ± 0.016(2)	**0.610 ± 0.019(1)**
yeast	0.628 ± 0.006(8)	0.632 ± 0.014(6)	0.637 ± 0.010(4)	0.631 ± 0.005(7)	0.633 ± 0.008(5)	0.640 ± 0.007(2)	0.637 ± 0.004(3)	**0.641 ± 0.011(1)**
health	0.637 ± 0.011(5)	0.631 ± 0.018(6)	0.362 ± 0.016(8)	0.591 ± 0.006(7)	0.637 ± 0.008(4)	**0.642 ± 0.002(1)**	0.639 ± 0.008(3)	0.641 ± 0.014(2)
core15k	0.034 ± 0.007(7)	0.115 ± 0.005(5)	0.027 ± 0.005(8)	0.101 ± 0.008(6)	0.238 ± 0.011(4)	0.241 ± 0.009(2)	0.240 ± 0.011(3)	**0.261 ± 0.005(1)**
arts	0.347 ± 0.011(7)	0.386 ± 0.015(5)	0.149 ± 0.006(8)	0.360 ± 0.009(6)	0.445 ± 0.015(2)	**0.446 ± 0.010(1)**	0.443 ± 0.008(3)	0.443 ± 0.009(4)
education	0.366 ± 0.015(7)	0.401 ± 0.011(5)	0.335 ± 0.016(8)	0.398 ± 0.021(6)	0.456 ± 0.011(4)	0.469 ± 0.005(2)	0.463 ± 0.011(3)	**0.476 ± 0.012(1)**
corel16k001	0.023 ± 0.005(7)	0.131 ± 0.019(5)	0.009 ± 0.001(8)	0.069 ± 0.002(6)	0.252 ± 0.007(3)	0.232 ± 0.008(4)	**0.253 ± 0.002(1)**	0.252 ± 0.004(2)
Ave.rank	6.9	5.6	6.8	6.0	3.7	2.5	2.5	**1.6**

comparison algorithms. Of course, we can get an appropriate value for the threshold of every algorithm with the use of a tuning phase. In this paper, we mainly study the effect of label correlations and label-specific features for multi-label classification, so we will learn the threshold in the next study.

4.4 Results of multi-label classification

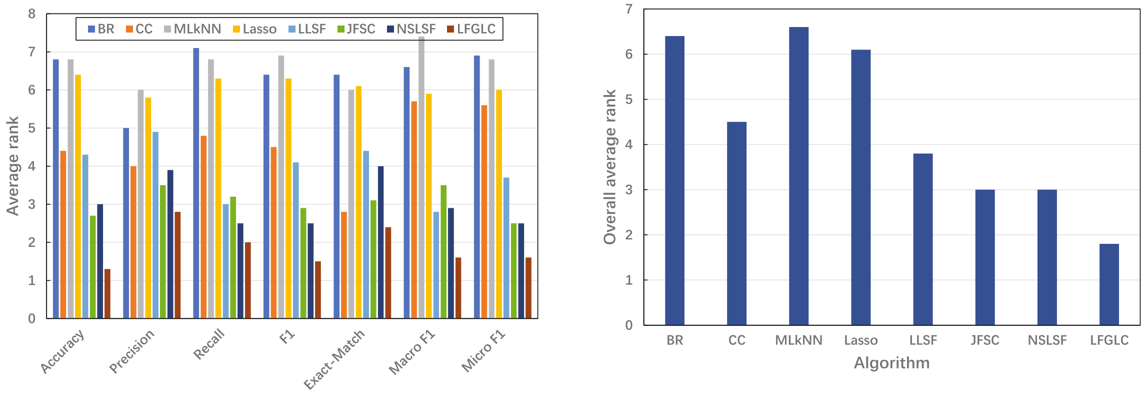
The experiment used 5-fold cross-validation on each data set to evaluate the performance of multi-label classification. The average results of each algorithm on 12 data sets with 7 evaluation metrics are summarized in Tables 2 and 3. The best result in a row is bolded. "↑" after a metric denotes that a larger value indicates better performance. For each evaluation metric, an "Ave.rank" row reports the average rank value over all data sets for each algorithm. A smaller rank indicates better performance. To more intuitively reflect the average rank of these algorithms, the average rank and overall average rank are depicted in Fig. 2. According to the experimental results in Tables 2 and 3, the observations are summarized as follows:

1. The proposed algorithm obviously outperforms the first-order algorithms (BR, MLkNN, Lasso) on all evaluation metrics, perhaps because LFGLC considers label correlation in multi-label classification, which is different from these first-order algorithms. Hence the consideration of label correlation can effectively improve multi-label classification performance.
2. Second-order (LLSF, JSFC, NSLSF) and high-order (CC) algorithms considere global label correlation but neglect local label correlation. LFGLC outperforms these algorithms, which indicates the effectiveness of local label correlation for multi-label classification. Compared with similar algorithms (LLSF, JSFC, LFGLC), the proposed algorithm can obtain more suitable regression parameters for classification.
3. On these evaluation metrics, LFGLC statistically performs better on *Accuracy*, *Precision*, *Recall*, F_1, *Exact-Match*, *Macro* F_1 and *Micro* F_1 over all data sets. This validates the superiority of the proposed algorithm. It is worth mentioning that *Precision* and *Recall* are generally contradictory. Because LFGLC considers the possibility that each instance may have labels similar to its neighbors, there will be more labels appearing in the prediction. Hence it performed better at *Recall*.

Figure 2 shows the overall average rank of algorithms, the order of all algorithms can be ranked as LFGLC$\succ$ NSLSF$\succ$ JFSC$\succ$ LLSF$\succ$ CC$\succ$ Lasso$\succ$ BR$\succ$ MLkNN. In summary, the proposed algorithm performs competitively in multi-label classification against other comparison algorithms.

To analyze the statistical performance among these algorithms systematically, Friedman test [45] was conducted here. Table 4 summarizes the Friedman statistics F_F and the critical value for each evaluation metric. This shows that the null hypothesis of equivalent performance among these comparison algorithms is rejected at significance level $\alpha = 0.10$ for each evaluation metric.

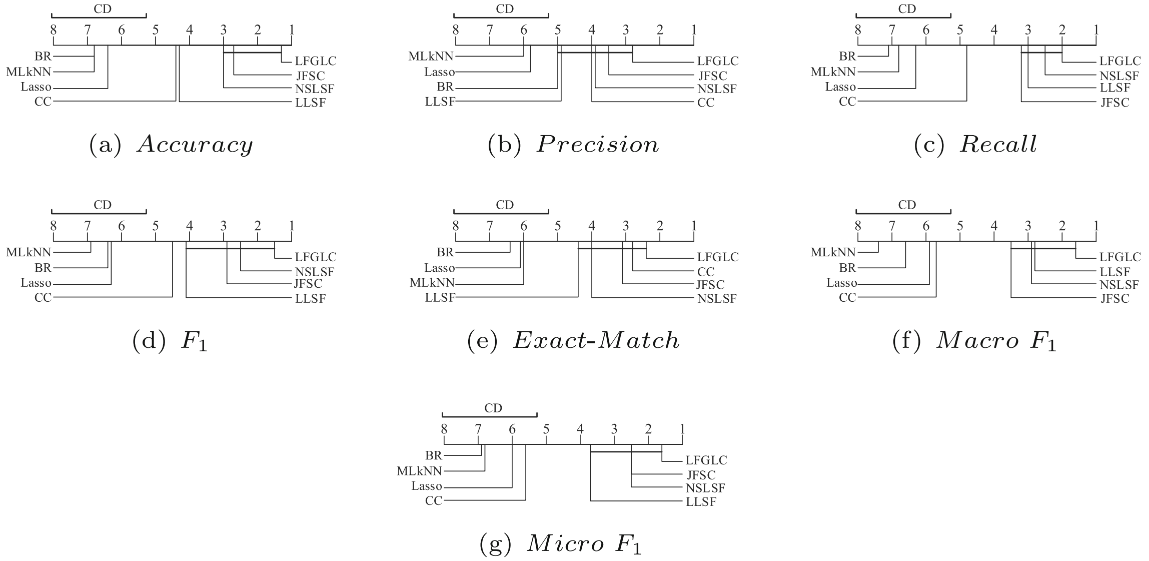
To analyze the relative performance among these algorithms, the post-hoc Nemenyi test [45] was conducted and LFGLC is treated as the control algorithm. The performance between control algorithm and one comparison algorithm will be significantly different if their average ranks differ by at least one CD (CD = 2.780 in this paper). Figure 3 shows the CD diagrams on each evaluation metric. In each subfigure, any comparison algorithm whose average rank is within one CD to that of LFGLC is connected, and otherwise it is considered to have significantly different performance against LFGLC.



(a) Average rank

(b) Overall average rank

Fig. 2 Results of the rank of all algorithms. Note: For ease of reading and understanding of these images, please read the corresponding online version of the paper (colors have been retained)



(a) *Accuracy*

(b) *Precision*

(c) *Recall*

(d) F_1

(e) *Exact-Match*

(f) *Macro* F_1

(g) *Micro* F_1

Fig. 3 Nemenyi test. The performance of those algorithms which are not significantly different from LFGLC are connected (CD=2.780 at 0.10 significance level)

4.5 Results of multi-label feature selection

To evaluate the performance of feature selection for multi-label learning, the proposed algorithm was compared with 4 feature selection based label-specific features algorithms (Lasso, LLSF, JSFC, NSLSF) on 5 multi-label data sets (cal500, emotions, medical, image, education) because of the space limitation. For each label, some features are selected from the original features according to the corresponding top weights from regression coefficients.

In the experiment, 5-fold cross-validation was conducted on each data set to evaluate the results in terms of *Accuracy*, F_1, *Exact-Match*, *Macro* F_1 and *Micro* F_1 for all algorithms. Because F_1 is the harmonic mean of *Precision* and *Recall*, for simplicity, we only select F_1 to evaluate the performance. The parameters of these algorithms are set the same as for multi-label classification. The top {10%, 20%, ..., 50%} of the original features are taken as the selected features. LIBSVM [44] with a linear kernel is employed as the base classifier for all algorithms. Figure 4 displays the average results of algorithms over each data set, and using "ALL" as a baseline, without selecting from the original features. According to the experimental results, the following observations can be made:

1. All feature selection algorithms generally outperform the baseline "ALL", which indicates that feature selection can effectively improve the performance of multi-label classification to some extent.
2. Label-specific features learned from LFGLC generally perform better than other comparison algorithms. Specifically, Lasso conducts feature selection without considering label correlation, LLSF, JFSC and NSLSF learn label-specific features with global label correlation. The results indicate that considering label correlation can be useful for feature selection and considering both global and local label correlation can further improve the performance of feature selection.
3. For different data sets, the performance of feature selection presents different change trends. The best performance for most data sets is obtained for some intermediate number of selected features, perhaps because few selected features will cause some important features missed, and a large number of selected features may introduce useless features, that degrade performance.

4.6 Global and local label correlation

To intuitively show the global and local label correlation learned by LFGLC, the global pairwise label correlation matrix and probabilities of labels in the neighborhood

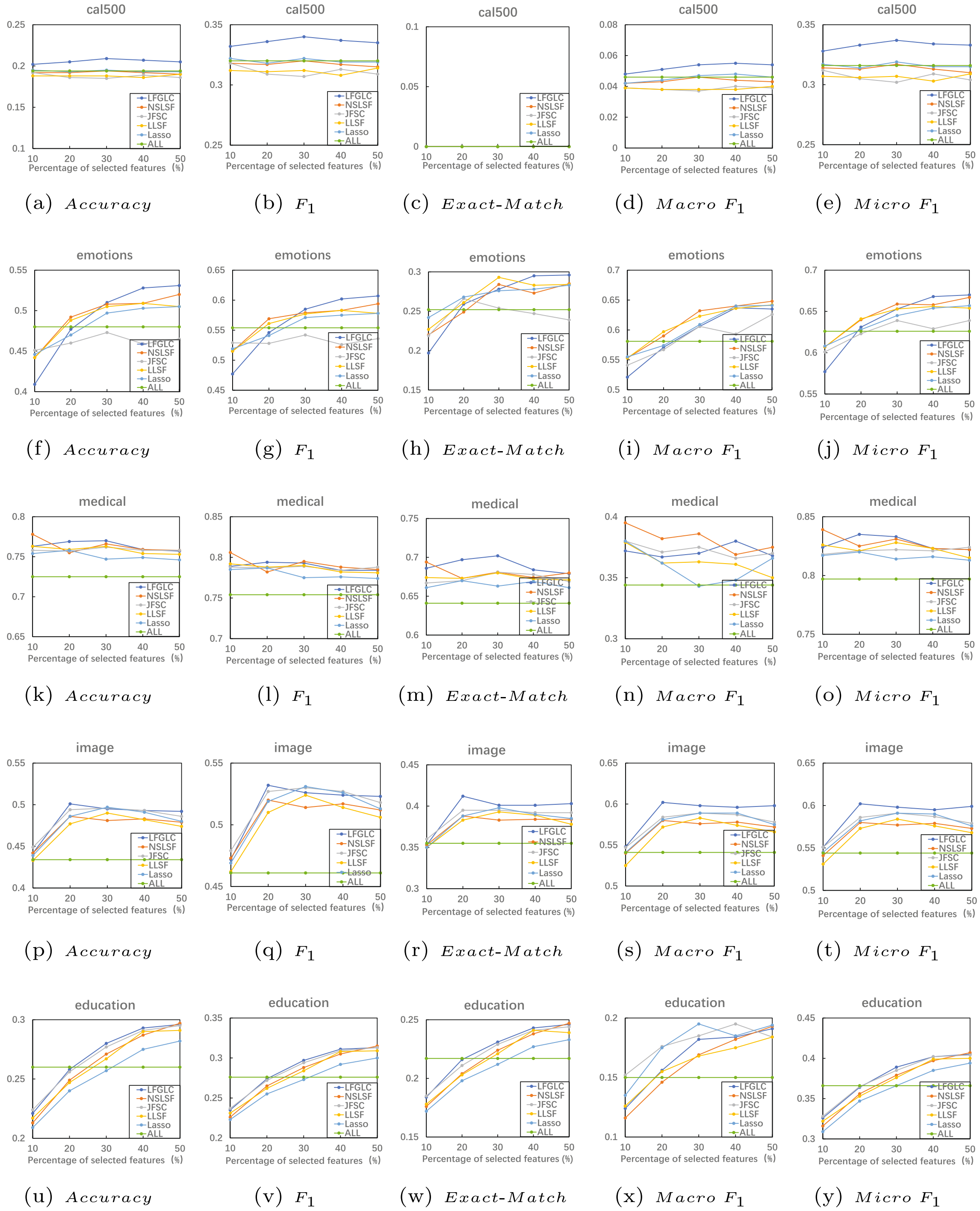
Table 4 Friedman statistics F_F and the critical value at 0.10 significance level in terms of each evaluation metric

Evaluation metric	F_F	Critical value
Accuracy	20.3646	1.7963
Precision	2.5683	
Recall	19.5354	
F_1	10.2219	
Exact-Match	3.7793	
Macro F_1	52.1148	
Micro F_1	21.9060	

learned from the Image data set are depicted in Fig. 5. We can observe from Fig. 5(a) that “mountains” is correlated with “trees”, and “sea” is correlated with “sunset” globally. In local label correlation (Fig. 5(b)), we randomly select 10 neighborhoods, whose label correlations are different. For example, “mountains” is correlated with “sea” in neighborhood 7, but “sea” is correlated with “sunset” in neighborhood 8. These further illustrate the complexity of label correlation.

4.7 Parameter sensitivity analysis

The objective function of LFGLC has several terms, whose contributions are controlled by these parameters (α, β,



(a) *Accuracy* (b) F_1 (c) *Exact-Match* (d) *Macro* F_1 (e) *Micro* F_1

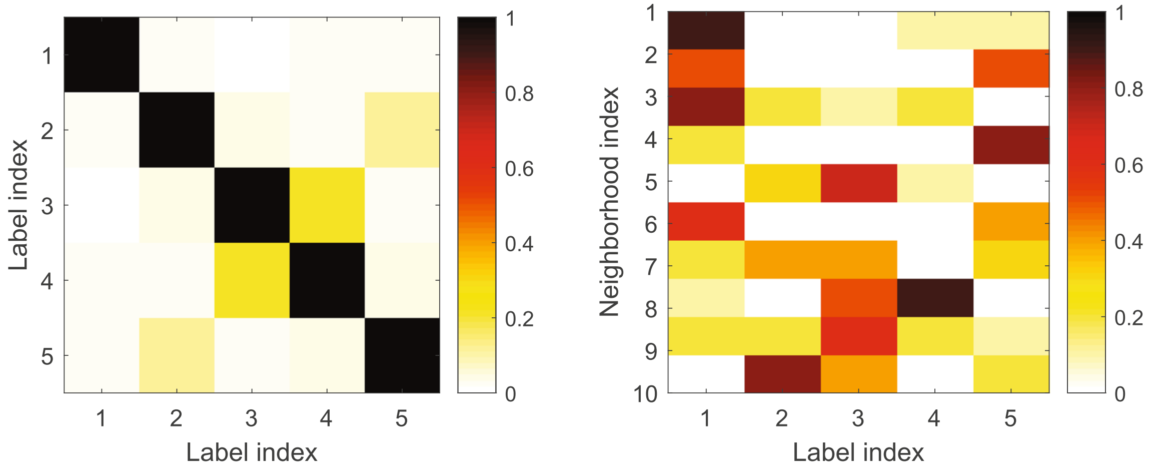
(f) *Accuracy* (g) F_1 (h) *Exact-Match* (i) *Macro* F_1 (j) *Micro* F_1

(k) *Accuracy* (l) F_1 (m) *Exact-Match* (n) *Macro* F_1 (o) *Micro* F_1

(p) *Accuracy* (q) F_1 (r) *Exact-Match* (s) *Macro* F_1 (t) *Micro* F_1

(u) *Accuracy* (v) F_1 (w) *Exact-Match* (x) *Macro* F_1 (y) *Micro* F_1

Fig. 4 Results of the performance of feature selection. Note: For ease of reading and understanding of these images, please read the corresponding online version of the paper (colors have been retained)



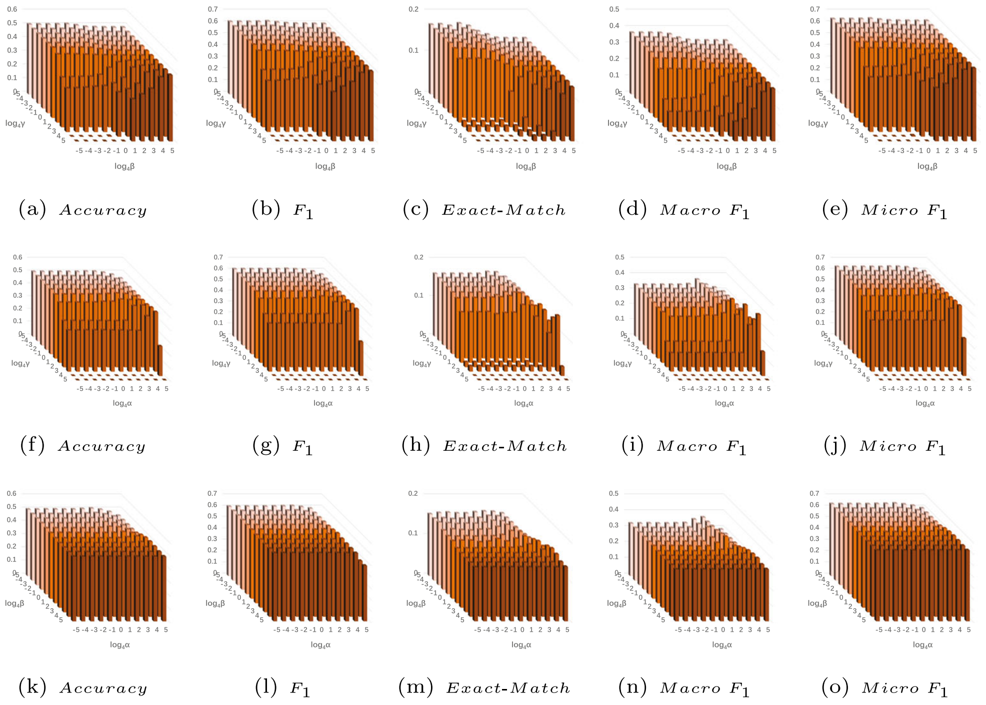
(a) Global label correlation.

(b) Local label correlation.

Fig. 5 Global and local label correlation on Image data set. The label index from small to large represents "desert", "mountains", "sea", "sunset" and "trees". Note: For ease of reading and understanding of these images, please read the corresponding online version of the paper (colors have been retained)

γ). The performance of LFGLC will be affected when the values of these parameters are changed. We conducted parameter sensitivity analysis of LFGLC on the yeast data set. Parameters corresponding to the best performance are first selected. Then the value of one parameter is fixed, and the values of the other two parameters are varied in $\{4^{-5}, 4^{-4}, \ldots, 4^{5}\}$.

Figure 6 shows the average results of 5-fold cross-validation in terms of *Accuracy*, F_1, *Exact-Match*, *Macro* F_1, *Micro* F_1. It can be seen that the best performance of *Accuracy*, F_1 and *Micro* F_1 is mostly obtained at the endpoint of the coordinate plane. But for evaluation metric *Exact-Match* and *Macro* F_1, some intermediate values of parameters achieve the best performance. Experimental results show that the performance of LFGLC is sensitive to parameters change in some intervals, and metrics generally achieve the best performance with different parameter values. Thus, to obtain the best performance on a certain data set, we suggest finding parameter values by searching on the validation set. Searching parameters will obviously cost much time especially for large-scale data sets.



(a) *Accuracy* (b) F_1 (c) *Exact-Match* (d) *Macro* F_1 (e) *Micro* F_1

(f) *Accuracy* (g) F_1 (h) *Exact-Match* (i) *Macro* F_1 (j) *Micro* F_1

(k) *Accuracy* (l) F_1 (m) *Exact-Match* (n) *Macro* F_1 (o) *Micro* F_1

Fig. 6 Parameter sensitivity analysis of LFGLC. Note: For ease of reading and understanding of these images, please read the corresponding online version of the paper (colors have been retained)

5 Conclusion

In this paper, we propose a new label-specific feature selection and multi-label classification algorithm LFGLC, which considers the complexity of real-world label correlation. Both global and local label correlation are taken into account to learn more discriminative label-specific features. For each label, label-specific features are selected from original features according to the nonzero regression coefficients. Experimental results show that combining global and local label correlation can be useful for multi-label learning. The proposed algorithm achieves a competitive performance against several algorithms in multi-label classification and feature selection. Considering that correlations between labels are not equal, we will try to find more compatible label correlation for multi-label learning in our future work.

Acknowledgements This research was supported by the natural Science Foundation of Fujian Province of China (No. 2021J011187).

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Lin J, Cai Q, Lin M (2021) Multi-label classification of Fundus images with graph convolutional network and self-supervised learning. IEEE Signal Process Lett 28:454–458
2. Huang X, Chen B, Xiao L, Yu J, Jing L (2021) Label-aware document representation via hybrid attention for extreme multi-label text classification. Neural Process Lett, pp 1–17
3. Wen J, Wei L, Zhou W, Han J, Guo T (2020) GCN-IA: user profile based on graph convolutional network with implicit association labels. In: Conference on computational science. pp 355–364
4. Sun Z, Zhang J, Dai L, Li C, Zhou C, Xin J, Li S (2019) Mutual information based multi-label feature selection via constrained convex optimization. Neurocomputing 329:447–456
5. Bayati H, Dowlatshahi M, Paniri M (2020) MLPSO: a filter multi-label feature selection based on particle swarm optimization. In: Conference on computer society of Iran pp 1–6
6. Zhang J, Luo Z, Li C, Zhou C, Li S (2019) Manifold regularized discriminative feature selection for multi-label learning. Pattern Recognit 95:136–150
7. Gonzalez-Lopez J, Ventura S, Cano A (2020) Distributed multi-label feature selection using individual mutual information measures. Knowl-Based Syst, p 188
8. Gonzalez-Lopez J, Ventura S, Cano A (2020) Distributed selection of continuous features in multilabel classification using mutual information. IEEE Trans on Neural Netw Learn Syst 31(7):2280–2293
9. Alalga A, Benabdeslem K, Taleb N (2015) Soft-constrained Laplacian score for semi-supervised multi-label feature selection. Knowl Inf Syst 47(1):75–98
10. Huang R, Jiang W, Sun G (2018) Manifold-based constraint Laplacian score for multi-label feature selection. Pattern Recognit Lett 112:346–352
11. Zhang ML, Wu L (2015) Lift: multi-label learning with label-specific features. IEEE Trans Pattern Anal Mach Intell 37(1):107–120
12. Guan Y, Li W, Zhang B, Han B, Ji M (2020) Multi-label classification by formulating label-specific features from simultaneous instance level and feature level. Applied Intell 9:1–16
13. Guo Y, Chung F, Li G, Wang J, Gee JC (2019) leveraging Label-specific discriminant mapping features for multi-label learning. ACM Trans Knowl Discovery Data 13(2):1–23
14. Weng W, Lin Y, Wu S, Li Y, Kang Y (2018) Multi-label learning based on label specific features and local pairwise label correlation. Neurocomputing 273:385–394
15. Huang J, Li GR, Wang SH, Xue Z, Huang QM (2017) Multi-Label Classification by exploiting local positive and negative pairwise label correlation. Neurocomputing 257:164–174
16. Huang R, Kang L (2021) Local positive and negative label correlation analysis with label awareness for multi-label classification. Int J Mach Learn Cybern, pp 1–14
17. Cheng Z, Zeng Z (2020) Joint label-specific features and label correlation for multi-label learning with missing label. Applied Intell 50(11):4029–4049
18. Bao J, Wang Y, Cheng Y (2021) Asymmetry label correlation for multi-label learning. Applied Intell, pp 1–13
19. Che X, Chen D, Mi J (2021) Feature distribution-based label correlation in multi-label classification. Int J Mach Learn Cybern, pp 1–15
20. Read J, Pfahringer B, Holmes G, Frank E (2011) Classifier chains for multi-label classification. Mach Learn 85:333–359
21. Che X, Chen D, Mi J (2020) A novel approach for learning label correlation with application to feature selection of multi-label data. Inf Sci 512:795–812
22. Li Q, Peng X, Qiao Y, Peng Q (2020) Learning label correlations for multi-label image recognition with graph networks. Pattern Recognit Lett 138:378–384
23. Ma J, Chiu B, Chow T (2020) Multilabel classification with group-based mapping: a framework with local feature selection and local label correlation. IEEE Trans Cybern
24. Nan G, Li Q, Dou R, Liu J (2018) Local positive and negative correlation-based k-labelsets for multi-label classification. Neurocomputing 318:90–101
25. Xiao J, Tang S (2020) Joint Learning of Binary Classifiers and Pairwise Label Correlations for Multi-label Image Classification. In: IEEE conference on multimedia information processing and retrieval. pp 25–30
26. Li YK, Zhang ML, Geng X (2015) Leveraging implicit relative labeling-importance information for effective multi-label learning. In: IEEE international conference on data mining. pp 251–260
27. Boutell MR, Luo J, Shen X, Brown CM (2014) Learning multi-label scene classification. Pattern Recognit 37:1757–1771
28. Zhang ML, Zhou ZH (2007) Ml-knn: a lazy learning approach to multi-label learning. Pattern Recognit 40:2038–2048
29. Wu G, Tian Y, Liu D (2018) Cost-sensitive multi-label learning with positive and negative label pairwise correlations. Neural Netw 108:411–423
30. Xu H, Xu L (2017) Multi-label feature selection algorithm based on label pairwise ranking comparison transformation. In: International joint conference on neural networks. pp 1210–1217
31. Zhang Y, Zhao T, Miao D, Pedrycz W (2021) Granular multilabel batch active learning with pairwise label correlation. IEEE Trans on Systems, Man, and Cybern
32. Wang R, Ye S, Li K, Kwong S (2021) Bayesian network based label correlation analysis for multi-label classifier chain. Inf Sci 554:256–275
33. He ZF, Yang M, Gao Y, Liu HD, Yin Y (2019) Joint multi-label classification and label correlations with missing labels and feature selection. Knowl-Based Syst 163:145–158

34. Zhu Y, Kwok JT, Zhou ZH (2018) Multi-Label Learning with global and local label correlation. IEEE Trans Knowl Data Eng 30(6):1081–1094
35. Yan Y, Li S, Xiao Z, Wang A, Li Z, Zhang J (2018) k-Labelsets for Multimedia Classification with Global and Local Label Correlation. In: International conference on multimedia Mmodeling. pp 177–188
36. Huang J, Li GR, Huang QM, Wu XD (2015) Learning label specific features for multi-label classification. In: IEEE international conference on data mining. pp 181–190
37. Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surveys 31(3):264–323
38. Wei XY, Yu ZW, Zhang CQ, Hu QH (2018) Ensemble of label specific features for multi-label classification. In: IEEE international conference on multimedia and expo. pp 1–6
39. Huang J, Xu L, Qian K, Wang J, Yamanishi K (2021) Multi-label learning with missing and completely unobserved labels. IEEE Trans Knowl Data Eng 35:1061–1086
40. Weng W, Chen YN, Chen CL, Wu SX, Liu JH (2020) Non-sparse label specific features selection for multi-label classification. Neurocomputing 377:85–94
41. Huang J, Li GR, Huang QM, Wu XD (2018) Joint feature selection and classification for multilabel learning. IEEE Trans Cybern 48(3):876–889
42. Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. Siam J Imaging Sci 2(1):183–202
43. Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc 58(1):267–288
44. Chang CC, Lin CJ (2011) LIBSVM: A library for support vectormachines. ACM Trans Intell Syst Technol 2(3):1–27
45. Demiar J, Schuurmans D (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7(1):1–30

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Wei Weng is currently an associate professor with the School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, China. His research interests include machine learning, artificial intelligence and deep learning for graph representation.

Bowen Wei is currently a M.E. student with the School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, China. His research interests include multi-label learning, machine learning and data mining.

Wen Ke is currently a M.E. student with the School of Computer and Information Engineering, Xiamen University of Technology, Xiamen, China. His research interests include multi-label learning, deep learning and natural language processing.

Yuling Fan is currently pursuing his Ph.D. degree in Department of Automation, School of Aerospace Engineering, Xiamen University, Xiamen, China. His research interests include data mining, machine learning and swarm intelligence.

Jinbo Wang is an assistant professor in Department of Statistics and Data Science, School of Economics, Xiamen University, Xiamen, China. His research interests include data mining, big data and statistics.

Yuwen Li is currently a lecture at School of Instrument Science and Engineering, Southeast University, Nanjing, China. Her research interests include data mining, and granular computing, machine learning and big data processing for physiological signals.