Graph-Margin Based Multi-label Feature Selection

Peng Yan and Yun Li(✉)

School of Computer Science and Technology,
Nanjing University of Posts and Telecommunications,
Wenyuanlu 9, Nanjing 210023, China
yanpeng9008@hotmail.com, liyun@njupt.edu.cn

Abstract. Since instances in multi-label problems are associated with several labels simultaneously, most traditional feature selection algorithms for single label problems are inapplicable. Therefore, new criteria to evaluate features and new methods to model label correlations are needed. In this paper, we adopt the graph model to capture the label correlation, and propose a feature selection algorithm for multi-label problems according to the graph combining with the large margin theory. The proposed multi-label feature selection algorithm GMBA can efficiently utilize the high order label correlation. Experiments on real world data sets demonstrate the effectiveness of the proposed method. The codes of the experiment of this paper are available at https://github.com/Faustus-/ECML2016-GMBA.

Keywords: Feature selection · Multi-label learning · Graph · Margin

1 Introduction

Multi-label learning studies the problem in which each instance is associated with a set of labels simultaneously. It usually occurs in text categorization, automatic annotation and bioinformatics, etc. [24]. For example, each music in emotions [15] data set can be associated with at most six different emotion tags simultaneously. A straightforward method to solve the multi-label problem is to decompose the problem into a series of single label binary classification problems, such as Binary Relevance [2] and ML-kNN [23]. However, this strategy neglects the label correlation which is usually helpful for improving the performance of a multi-label learning algorithm. To complement this, various multi-label learning algorithms with the consideration of label correlation have been proposed, such as [4,7,8,11,13,19,22]. According to the utilization of label correlation, these algorithms can be divided into three orders [24]: (a) the first order algorithms predict labels for an unseen instance one by one. They are very simple while neglecting label correlation [2,23]. (b) the second order algorithms consider pairwise relation between labels, which usually leads to a label ranking problem [4,8]. (c) the high order algorithms capture more complex correlation between labels, but they are computationally expensive [7,11,13,19,22].

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part I, LNAI 9851, pp. 540–555, 2016.
DOI: 10.1007/978-3-319-46128-1_34

Similar to other machine learning tasks, multi-label learning also suffers from the curse of dimensionality. Redundant and irrelevant features make data intractable, resulting in unreliable model and degraded learning performance. Feature selection is an efficient and popular technique to reduce dimensionality. Several feature selection algorithms for the multi-label problem have been presented. For example, feature selection algorithms for multi-label naive bayes classifier and Rank-SVM classifier are introduced in [5,21] respectively. These multi-label feature selection algorithms belong to the wrapper model [14], which evaluates features according to predictive results of the specified learning algorithm, thus they share bias of the learning algorithm and it is prohibitively expensive to run for data with a large number of features. In [6], two classic single-label feature selection algorithms, F-Statistic and ReliefF, are extended to handle multi-label problems. These algorithms belong to the filter model [14], which evaluates features by measuring the statistics of a multi-label data set. Algorithms belonging to the filter model are independent of specified classifiers and more flexible than those belonging to the wrapper model.

In this paper, a graph-margin based multi-label feature selection algorithm (GMBA) is proposed. GMBA firstly describes multi-label data with a graph, which has good discrimination capability and shares similar expression capability to the hypergraph applied in [13,19]. Then, it measures features based on the graph combining with the large margin theory. Since GMBA evaluates features according to the graph derived from the training data, it is independent of a specified learning algorithm and belongs to the filter model. We will introduce GMBA in the following order. In Sect. 2, we define a similarity measure for multi-label instances and describe multi-label data by a graph. The discrimination capability and expression capability of the graph are also discussed in this section. In Sect. 3, we define a margin for multi-label data and derive GMBA depending on the graph combining with the margin. In addition, experimental results on real world data sets are reported in Sect. 4 and paper concludes in Sect. 5.

Notations. Before introducing the algorithm, we will give the notations in this paper. n, D and Q denote the number of training instances, the data dimensionality, and the number of labels, respectively. F^d denotes the dth feature and l^q denotes the qth label, where $1 \leq d \leq D$ and $1 \leq q \leq Q$. n_q denotes the number of training samples associated with l^q. $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ denotes the ith instance in the training data.

$\boldsymbol{x}_i = \left(x_i^1, ..., x_i^d, ..., x_i^D\right)$ denotes the features of the ith instance in the training data, where x_i^d denotes the dth component of the ith instance, or the ith instance has value x_i^d for F^d.

$\boldsymbol{x}^d = \left(x_1^d, ..., x_i^d, ..., x_n^d\right)^T$ denotes a feature vector of the dth feature. The superscript T means the transpose of a vector or matrix.

$\boldsymbol{y}_i = \left(y_i^1, ..., y_i^q, ..., y_i^Q\right)$ denotes the relationship between labels and the ith instance in the training data. If the ith instance is associated with l^q then $y_i^q = 1$, or $y_i^q = 0$. For single-label problems, there is a constraint that $|\boldsymbol{y}_i| = 1$, $1 \leq i \leq n$, where $|\cdot|$ denotes the 1-norm of the vector.

$s_{(i,i')}$ denotes the similarity between the ith and i'th instances.

$\mathbf{G} = (\mathbf{V}, \mathbf{E})$ denotes a graph, where $\mathbf{V}$ and $\mathbf{E}$ denote the vertex set and the edge set of the graph, respectively. $\boldsymbol{A}_G$ denotes the adjacent matrix of $\mathbf{G}$, $\boldsymbol{D}_G$ denotes the degree matrix of $\mathbf{G}$. $\boldsymbol{L}_G = \boldsymbol{D}_G - \boldsymbol{A}_G$ is the corresponding Laplacian matrix.

$[\pi]$ returns 1 if predicate π holds, and 0 otherwise.

$|\cdot|$ and $\|\cdot\|$ returns 1-norm and 2-norm respectively.

$\boldsymbol{\omega}$ is a weight vector of features.

2 Graph Model for Multi-label Data

2.1 Graph Definition

Graph is a widely used model for its powerful expression capability. For example, well-known page rank and image segmentation algorithm in [3] are based on the graph model. In this paper, we adopt the graph to capture the correlation between labels and instances for multi-label data.

Suppose, in a graph, each vertex $v_i \in \mathbf{V}$ represents an instance and an edge $e_{(i,i')} \in \mathbf{E}$ connecting two vertexes denotes the similarity of the corresponding instances, then a simple undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$ can be built to model the correlation between instances. The key of building the graph depends on how one measures the instance similarity. For a single-label problem, the similarity between two instances $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$ are usually defined as Eq. 1 [25].

$$s_{single}(i, i') = \begin{cases} \frac{1}{n_q}, & y_i^q = y_{i'}^q = 1 \\ 0, & otherwise \end{cases} \tag{1}$$

which means that instances in the same class share the same similarity, while similarity between instances from different classes is 0. However, when it comes to multi-label problems, an instance is associated with several labels (classes) simultaneously and it is ambiguous to compare the belongingness of two different instances. Therefore, Eq. (1) is not suitable when solving multi-label problems and we define Eq. 2 to measure the similarity between two multi-label instances.

$$s_{multi}(i, i') = \begin{cases} \frac{\sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \wedge y_{i'}^q = 1]}{\sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \vee y_{i'}^q = 1]}, & \sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \vee y_{i'}^q = 1] \neq 0 \\ 0, & otherwise \end{cases} \tag{2}$$

In Eq. 2, the numerator counts the labels two instances shared, and the denominator counts the labels at least one of the two instances associated with. n_q is the number of training samples associated with l_q and it is applied as a weight to tune the importance of different labels. Equation 2 is a variation of the Jaccard similarity, which measures the ratio of the size of intersection and the size of union for two sets. Then, the multi-label data can be represented as a graph using an adjacent matrix $\boldsymbol{A}_G$ definded in Eq. 3, where $A_G(i, i')$ is the element in the ith row, i'th column of $\boldsymbol{A}_G$.

$$A_G(i, i') = \begin{cases} s_{multi}(i, i'), & i \neq i' \\ 0, & otherwise \end{cases} \tag{3}$$

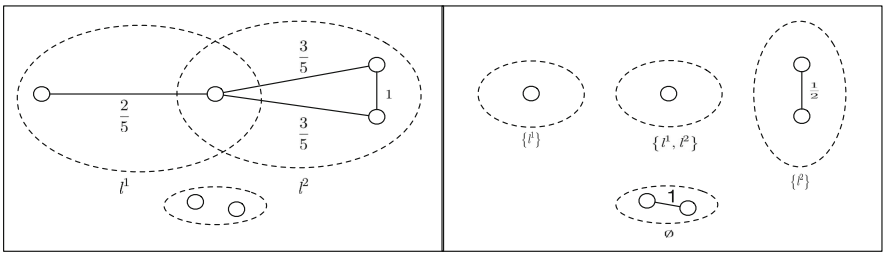

Fig. 1. (a) Our proposed graph for multi-label data, (b) The graph for multi-class data transformed from the multi-label data. The edges in graphs are denoted with solid lines and circles are vertexes. The fractions on the edges represent the similarity weight. Circles fallen in the same ellipse (dash line) represent instances associated with the same label/class. The circle fallen in the intersection of two ellipses means the instance is associated with two labels simultaneously. And the two instances associated with no labels are put in the ellipse below.

2.2 Discrimination Capability

To explain the discrimination capability of the proposed graph, an example is presented below. Assuming that there are Q different labels, we have $\boldsymbol{y}_i \in \{0,1\}^Q$. Without loss of generality, we set $Q = 2$ and two labels are named l^1 and l^2. We also assume that a multi-label training data set consists of one instance associated with l^1, two instances associated with l^2, one instance associated with l^1 and l^2 simultaneously and two instances associated with no labels. The proposed graph to describe these instances is given in Fig. 1(a). Then, if one can split the graph into different parts (such as the ellipses of dash line), instances associated with different labels will be discriminated. Hence multi-label instances are discriminable in the proposed graph. Moreover, some off the shelf algorithms can be applied to finish this task, such as normalized cut [12], ration cut [18], etc.

In addition, the discrimination capability of the proposed graph is similar to the one derived from label power set algorithms as in [16,17], while the proposed graph is smoother and can capture label correlation. More specifically, a label power set algorithm usually transforms a multi-label problem into a multi-class problem in which each class corresponds to a label power set. For the multi-label problem mentioned above, a label power set algorithm will transform it into a multi-class problem with 4 different classes: $\emptyset$, $\{l^1\}$, $\{l^2\}$ and $\{l^1, l^2\}$, and each instance is associated with one class. Since a multi-class problem belongs to the single-label learning problem, the similarity between instances can be measured by Eq. 1. The resulting graph is shown in Fig. 1(b), which includes 4 unconnected subgraphs. The partitions of the graph (ellipses of dash line) are similar to the ones in Fig. 1(a), hence they have similar discrimination capability. However, in multi-class problems, the similarity between instances from different classes is 0, and there are no edges connecting them, such as the instance belonging to

the class $\{l^1\}$ and the one belonging to the class $\{l^1, l^2\}$ in Fig. 1(b). Although these instances actually share some labels in common, such as the label l^1 for the class $\{l^1\}$ and the class $\{l^1, l^2\}$, the correlation is not considered by the graph in Fig. 1(b). On the contrary, such kind of correlation is considered in our graph as in Fig. 1(a) through the edges weight between 0 and 1. Therefore, the proposed graph for a multi-label problem is smoother than the graph for a multi-class problem transformed from a multi-label problem in [16,17] and can capture label correlation.

2.3 Expression Capability

Though the proposed graph in Sect. 2.1 is a simple-graph, it has similar expression capability to a hypergraph, which has been successfully applied to capture high order label correlation in [13,19].

Different from edges in a simple-graph, an edge, which is called hyperedge, in a hypergraph connects more than two vertexes simultaneously. Hence multi-label data can be described by a hypergraph as follows: in a hypergraph $\mathbf{G}_H = (\mathbf{V}_H, \mathbf{E}_H)$, each vertex $v_i \in \mathbf{V}_H$ corresponds to an instance in the multi-label data set, each hyperedge $e_q \in \mathbf{E}_H$ is a subset of $\mathbf{V}_H$, where $e_q = \{v_i \mid y_i^q = 1, 1 \leq i \leq n\}$. The degree of each hyperedge $d(e_q)$ is defined as the number of vertexes on that hyperedge, namely n_q, and we may set the weight of a edge, $w(e_q)$, equals to its degree.

If we apply Clique Expansion [1,13,19] to expand the hypergraph above, we obtain a simple-graph $\mathbf{G}_C = (\mathbf{V}_C, \mathbf{E}_C)$, where $\mathbf{V}_C = \mathbf{V}_H$ and $\mathbf{E}_C = \{e_{(i,i')} \mid v_i \in e_q \wedge v_{i'} \in e_q, e_q \in \mathbf{E}_H\}$. The weight of $e_{(i,i')}$ is defined as Eq. 4.

$$w\left(e_{(i,i')}\right) = \sum_{v_i \in e_q \wedge v_{i'} \in e_q, e_q \in \mathbf{E}_H} w(e_q) = \sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \wedge y_{i'}^q = 1] \tag{4}$$

Normalizing it to obtain Eq. 5, we find that Eq. 5 is the same to the similarity defined in Eq. 2

$$\hat{w}\left(e_{(i,i')}\right) = \frac{\sum_{v_i \in e_q \wedge v_{i'} \in e_q, e_q \in \mathbf{E}_H} w(e_q)}{\sum_{v_i \in e_q \vee v_{i'} \in e_q, e_q \in \mathbf{E}_H} w(e_q)} = \frac{\sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \wedge y_{i'}^q = 1]}{\sum_{q=1}^{Q} n_q \cdot [y_i^q = 1 \vee y_{i'}^q = 1]} \tag{5}$$

Thus our proposed graph is the same to the simple-graph expanded from a hypergraph by Clique Expansion. According to [1,13], both the hypergraph and the expanded simple-graph, as well as the proposed graph, can capture similar high order correlation and therefore they share similar expression capability for multi-label data.

3 Graph-Margin Based Multi-label Feature Selection (GMBA)

In Sect. 2, we propose a discriminative graph to describe multi-label data. According to the similarity measure defined in Eq. 2, the graph reflects the relations of data in label space. However, these relations in label space are usually

different from the one in feature space. We will illustrate this case in Fig. 2(a) and (b). For an instance denoted by star in Fig. 2(a), its several nearest neighbors in label space are represented by squares. That is to say, the similarity measured by Eq. 2 between a square and the star is greater than a threshold s_{min}, and these squares are the closest instances to the star in the proposed graph as in Fig. 2(a). However, if we estimate similarities among instances in feature space, such as using a radial basis function, an instance represented by triangle could be more similar (closer) to the star than squares. This means that the graph built in feature space as in Fig. 2(b) is inconsistent with the graph in label space as in Fig. 2(a).

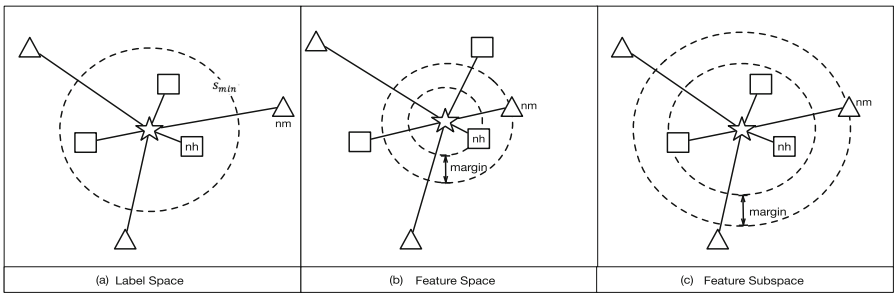

Fig. 2. A comparison of graphs built in different spaces. Each star, square or triangle represents an instance. The edges connect two different shapes denote the similarity between them. The shorter an edge is, the more similar two instances are. We omit the edges that do not connect with the star.

Furthermore, for classification problems, the target of a classifier is using features to divide instances into different classes, which means that we have to use features to predict the partitions of the graph in label space. Although the graph built in label space is discriminative as analyzed in Sect. 2.2, an inconsistent counterpart in feature space does not maintain its discrimination power and may lead to wrong partition. Thus we propose a multi-label feature selection algorithm GMBA, which will choose a subset of features that the graph built in this feature subspace, as in Fig. 2(c), is similar to the one in label space, as Fig. 2(a). In addition, a margin, as depicted in Fig. 2(c), is applied in GMBA to guarantee the generalization capability of the selected features.

3.1 Loss Function

To evaluate the inconsistency described above, we design a loss function based on margin. Firstly, we apply $sim\,(i)$ and $dissim\,(i)$ to represent the instance subsets similar and dissimilar to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in label space respectively. They are described in Eqs. 6 and 7.

$$sim\,(i) = \{(\boldsymbol{x}_{i'}, \boldsymbol{y}_{i'}) \mid s_{multi}\,(i, i') \geq s_{min}, 1 \leq i' \leq n \text{ and } i \neq i'\} \tag{6}$$

$$dissim\,(i) = \{(\boldsymbol{x}_{i'}, \boldsymbol{y}_{i'}) \mid s_{multi}\,(i, i') < s_{min}, 1 \leq i' \leq n \text{ and } i \neq i'\} \tag{7}$$

where s_{min} is a given threshold and $s_{multi}\,(i, i')$ is the similarity defined in Eq. 2. Then, the loss function is designed in Eq. 8 to evaluate the inconsistency between the graph in label space and the one in feature space for $(\boldsymbol{x}_i, \boldsymbol{y}_i)$.

$$Loss\,(i) = \sum_{i' \in neighbor(i)} s_{multi}\,(i, i') \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\right\|^2 + \lambda \sum_{i'' \in dissim(i)} \delta\,(i', i'') \tag{8}$$

where $neighbor\,(i)$ denotes a instance subset with k instances those are both nearest to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in feature space and belong to $sim\,(i)$. The first term of Eq. 8 penalizes large distance between $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ and its neighbors $(\boldsymbol{x}_{i'}, \boldsymbol{y}_{i'})$ in $neighbor\,(i)$. The second term $\delta\,(i', i'')$ is a penalty defined in Eq. 9 and λ is the tuning parameter.

$$\begin{aligned} &\delta\,(i', i'') = \\ &(s_{multi}\,(i, i') - s_{multi}\,(i, i'')) \cdot \max\left(0, m\,(i) + \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\right\|^2 - \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i''}\right\|^2\right) \end{aligned} \tag{9}$$

$\delta\left(i', i^{i''}\right)$ is the hinge loss penalizing $(\boldsymbol{x}_{i''}, \boldsymbol{y}_{i''})$, which is an instancec in $dissmiss\,(i)$ but closer to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ than $(\boldsymbol{x}_{i'}, \boldsymbol{y}_{i'})$ to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in feature space. The closer $(\boldsymbol{x}_{i''}, \boldsymbol{y}_{i''})$ to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in feature space and more dissimilar $(\boldsymbol{x}_{i''}, \boldsymbol{y}_{i''})$ to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in label space, the larger the penalty. $m\,(i)$ is the margin defined in Eq. 10, where $\mathrm{nh}\,(i)$ and $\mathrm{nm}\,(i)$ are the nearest instances from $sim\,(i)$ and $dissim\,(i)$ respectively to the $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in feature spaces.

$$m\,(i) = \left|\left\|\boldsymbol{x}_i - \boldsymbol{x}_{\mathrm{nh}(i)}\right\|^2 - \left\|\boldsymbol{x}_i - \boldsymbol{x}_{\mathrm{nm}(i)}\right\|^2\right| \tag{10}$$

We will illustrate the penalty defined Eq. 9 for the case depicted in Fig. 2(b). Assuming that the star represents $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, the margin $m\,(i)$ is the absolute value of the square Euclidean distance between the square marked nh and the star minus the square Euclidean distance between the triangle marked nm and the star. If the square Euclidean distance between any triangle and the star is smaller than the square Euclidean distance between a square and the star plus this margin, it will be penalized by Eq. 9.

3.2 Feature Ranking

Based on the loss function in Eq. 8, one can evaluate the inconsistency between the graph in label space and the one in feature space by summing up the loss of all training data as depicted in Eq. 11. The smaller Eq. 11 is, the more consistent two graphs are. In addition, for feature selection, it is key to find a feature subspace that minimize Eq. 11.

$$Loss\,(\mathbf{G}) = \sum_{i=1}^{n} Loss\,(i) \tag{11}$$

However, it suffers from the complexity of $O\left(2^D\right)$ to find the best subspace for Eq. 11. As a result, according to [9,10], we evaluate the fitness of features by a weight vector $\boldsymbol{\omega}$ and find the best $\boldsymbol{\omega}$ by gradient descent method. Specifically, searching for the best $\boldsymbol{\omega}$ can be formulated as Eq. 12

$$\min_{\boldsymbol{\omega}} Loss\left(\boldsymbol{\omega}, \mathbf{G}\right) = \min_{\boldsymbol{\omega}} \sum_{i=1}^{n} Loss\left(\boldsymbol{\omega}.i\right) \tag{12}$$

where

$$Loss\left(\boldsymbol{\omega}.i\right) = \sum_{i' \in neighbor(i)} s_{multi}\left(i, i'\right) \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\right\|_{\boldsymbol{\omega}}^2 + \lambda \sum_{i'' \in dissim(i)} \delta\left(\boldsymbol{\omega}, i', i''\right) \tag{13}$$

$$\begin{aligned} &\delta\left(\boldsymbol{\omega}, i', i''\right) \\ &= \left(s_{multi}\left(i, i'\right) - s_{multi}\left(i, i''\right)\right) \cdot \max\left(0, m\left(i\right) + \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\right\|_{\boldsymbol{\omega}}^2 - \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i''}\right\|_{\boldsymbol{\omega}}^2\right) \end{aligned} \tag{14}$$

and $\|\boldsymbol{z}\|_{\boldsymbol{\omega}} = \sqrt{\sum_{d=1}^{D} \left(\omega^d z^d\right)^2}$.
Then Eq. 12 can be solved by the gradient descent and the algorithm is summarized as follows.

Step 1: Initialize $\boldsymbol{\omega} = (1, 1, 1, ..., 1)$, and set the number of iterations I.
Step 2: For i =1, 2, ... , I.
(a) Pick up an instance $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, and find $sim(i)$ and $dissim(i)$ according to Eqs. 2, 6 and 7.
(b) Find k nearest instances to $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ in feature space from $sim(i)$ as $neighbor(i)$.
(c) Find nh(i) and nm(i) from $sim(i)$ and $dissim(i)$ respectively.
(d) Calculate $m(i)$ according to Eq. 10
(e) For d =1, 2, ... , D
$\nabla^d = 2\omega^d \sum_{i' \in neighbor(i)} s_{multi}\left(i, i'\right) \left\|x_i^d - x_{i'}^d\right\|^2 + \lambda \sum_{i' \in dissim(i)} \frac{\partial \delta\left(\boldsymbol{\omega}, i', i''\right)}{\partial \omega^d}$,
where $\frac{\partial \delta\left(\boldsymbol{\omega}, i', i''\right)}{\partial \omega^d}$ is the partial derivative of $\delta\left(\boldsymbol{\omega}, i', i''\right)$ given in Eqs. 15 and 16

$$\frac{\partial \delta\left(\boldsymbol{\omega}, i', i''\right)}{\partial \omega^d} = \begin{cases} 0, & m\left(i\right) + \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\right\|^2 < \left\|\boldsymbol{x}_i - \boldsymbol{x}_{i''}\right\|^2 \\ diff(d), & otherwise \end{cases} \tag{15}$$

$$diff(d) = 2\omega^d \left(s_{multi}\left(i, i'\right) - s_{multi}\left(i, i''\right)\right) \left(\left\|x_i^d - x_{i'}^d\right\|^2 - \left\|x_i^d - x_{i''}^d\right\|^2\right) \tag{16}$$

(f)$\boldsymbol{\omega} = \boldsymbol{\omega} - \beta \boldsymbol{\nabla} / \|\boldsymbol{\nabla}\|$, where β is a decay factor.
Step 3: Ranking features based on $\boldsymbol{\omega}$. The greater the ω^d, the better the F^d.

4 Experiments

To demonstrate the effectiveness of the proposed GMBA, we empirically compare the GMBA with the multi-label F-Statistic (MLFS)[6] and the multi-label ReliefF (MLRF) [6].

In addition, spectral feature selection framework (SPEC) [25] is an algorithm which selects features based on the graph structure for single label problems. It measures features according to Eq. 17.

$$\phi\left(F^d\right) = \left(\hat{\boldsymbol{x}}^d\right)^T \boldsymbol{\mathcal{L}}_{\boldsymbol{G}} \hat{\boldsymbol{x}}^d = \sum_{1 \le i, i' \le n} \frac{s_{single}\left(i, i'\right)}{\sqrt{d\left(i\right) d\left(i'\right)}} \left\| \hat{x}_i^d - \hat{x}_{i'}^d \right\|^2 \tag{17}$$

where $d(i)$ is the degree of vertex v_i, $\hat{\boldsymbol{x}}^d = \frac{\boldsymbol{D}^{\frac{1}{2}} \boldsymbol{x}^d}{\left\| \boldsymbol{D}^{\frac{1}{2}} \boldsymbol{x}^d \right\|}$ is the normalized feature vector and $\boldsymbol{\mathcal{L}}_{\boldsymbol{G}} = \boldsymbol{D}_G^{-\frac{1}{2}} \boldsymbol{L}_G \boldsymbol{D}_G^{-\frac{1}{2}}$ is the normalized Laplacian matrix. The smaller the Eq. 17, the better the F^d. We adapt it to multi-label problems by replacing the $s_{single}\left(i, i'\right)$ with the proposed similarity $s_{multi}\left(i, i'\right)$, so that it will select features consistent with the proposed graph structure for multi-label data.

4.1 Data Sets

Eight benchmark multi-label data sets from different domains are used for experiments, which are downloaded from MULAN[1]. Details about data sets are listed in Table 1. All numerical features are normalized with zero mean and unit variance in experiments. Features with variance 0 are eliminated.

Table 1. Summary of 8 benchmark data sets

Name	Instance	Features	Labels	Domain	Name	Instance	Features	Labels	Domain
bibtex	7395	1836	159	text	mediamill	43907	120	101	video
emotions	593	72	6	music	medical	978	1449	45	text
enron	1702	1001	53	text	scene	2407	294	6	image
genebase	662	1186	27	biology	yeast	2417	103	14	biology

4.2 Classifiers and Parameters

Binary Relevance [2] (1_{st} order algorithm) and Classifier Chain [11] (high order algorithm) are used as multi-label learning strategy respectively, 3-Nearest Neighbor (3-NN) classifier in scikit-learn[2] is applied as the base classifier. Number of neighbors for MLRF and $neighbor\,(i)$ in GMBA are set 3. The threshold s_{min} and tuning parameter λ are 1. The number of iterations I equals to the number of training data n. The decay factor β is 0.9. Experiments[3] are carried on under the environment of Python 2.7.

[1] http://mulan.sourceforge.net/.
[2] http://scikit-learn.org/stable/.
[3] Codes can be acquired at https://github.com/Faustus-/ECML2016-GMBA.

4.3 Evaluations

Three different measurements [24], i.e., Hamming loss (↓), micro (↑) and macro (↑) F1-Measure, are applied to validate the performance of the selected features for multi-label learning. (↓) denotes the smaller the better, while (↑) denotes the larger the better. Except for mediamill and bibtex, all results reported in this paper are the average of 10-cross validation. Since the big size of mediamill and bibtex, we randomly select 1800 instances and other 10 percent of total instances for training and testing respectively. The results reported are the average of 10 trials of experiments.

4.4 Results

Experimental results are shown in Figs. 3, 4, 5 and 6. For space limitation, we display the Hamming Loss for the bibtex, emotions, enron and genebase, macro F1-Measure metrics for mediamill, medical, scene and yeast when the multi-label learning strategy is Binary Relevance. We also display the Hamming Loss for mediamill, medical, scene and yeast, macro F1-Measure metrics for bibtex, emotions, enron and genebase when the multi-label learning strategy is Classifier Chain. Complete results of micro F1-Measure metrics are displayed in Figs. 5 and 6.

Experimental results show that features selected by proposed GMBA obtain better classifying performance than others in most cases. For emotions and scene data sets, all algorithms achieve similar performance, which might result from the fact that there are only 6 labels, causing a weak discrimination power of graphs built in the label space. In addition, GMBA and the adapted SPEC are suitable for more data sets than MLFS and MLRF, since the performance of MLFS and MLRF vary from different data sets.

5 Discussions and Conclusions

According to experimental results, GMBA performs better than other algorithms, and both GMBA and SPEC are suitable for more data sets than MLFS and MLRF. In addition, while GMBA and SPEC all aim to find a feature subset that the graph built in this subspace is consistent with the graph built in label space, GMBA is superior to the SPEC in most cases. This results from the margin we applied in GMBA, since a margin usually leads to better discrimination and generalization, such as LMNN in [20] and the classic SVM. More specifically, as illustrated in Fig. 2(c), similar instances are *pushed* close to each other and dissimilar instances are *pulled* away from them according to the margin. In this way, the margin makes features in this subspace become more discriminative.

In conclusion, based on the graph and the large margin theory, the proposed GMBA can capture high order label correlation and guarantee generalization capability. Experimental results on different real world data sets indicate the effectiveness and good performance of the proposed algorithm.

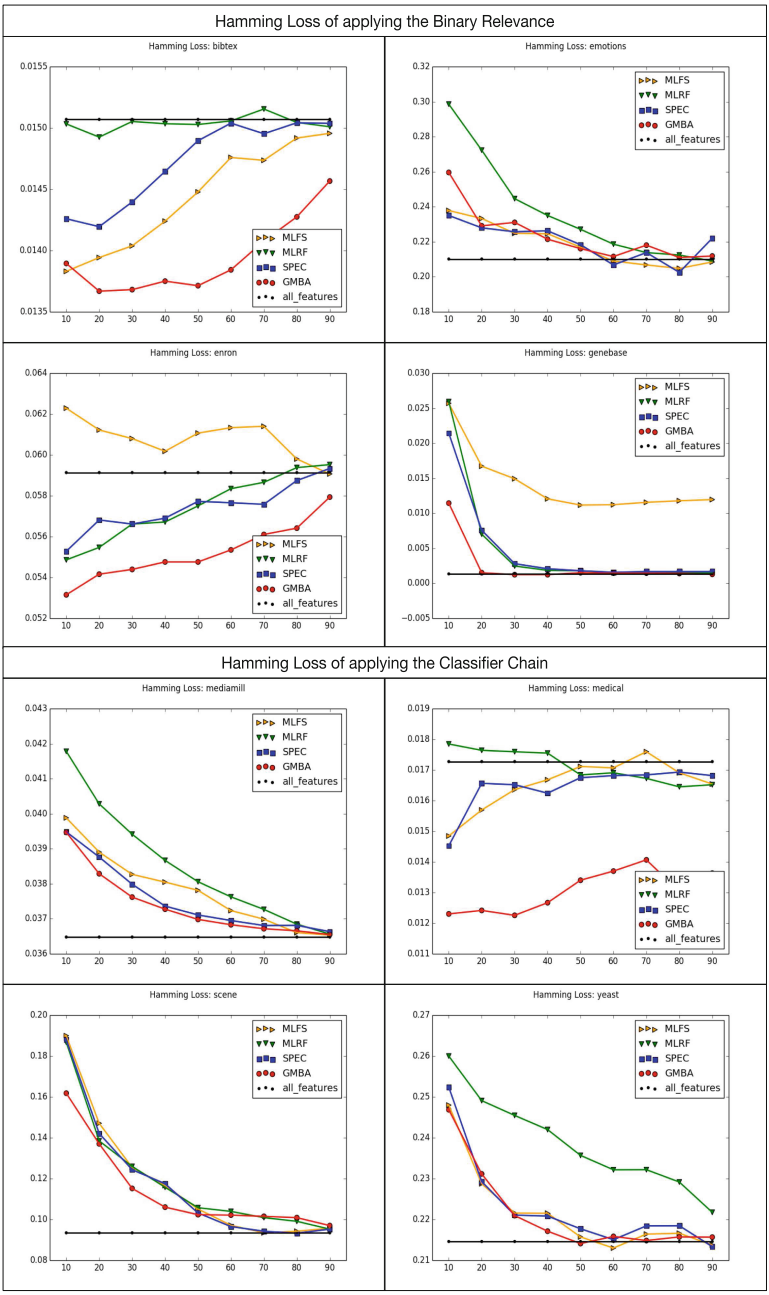

Fig. 3. Hamming loss (↓). The first 4 diagrams show the hamming loss of applying the Binary Relevance while the rest show the results from the Classifier Chain. Y-axis corresponds to different metrics and X-axis denotes the percentage of features selected. The horizontal lines are the results of classifying with all features

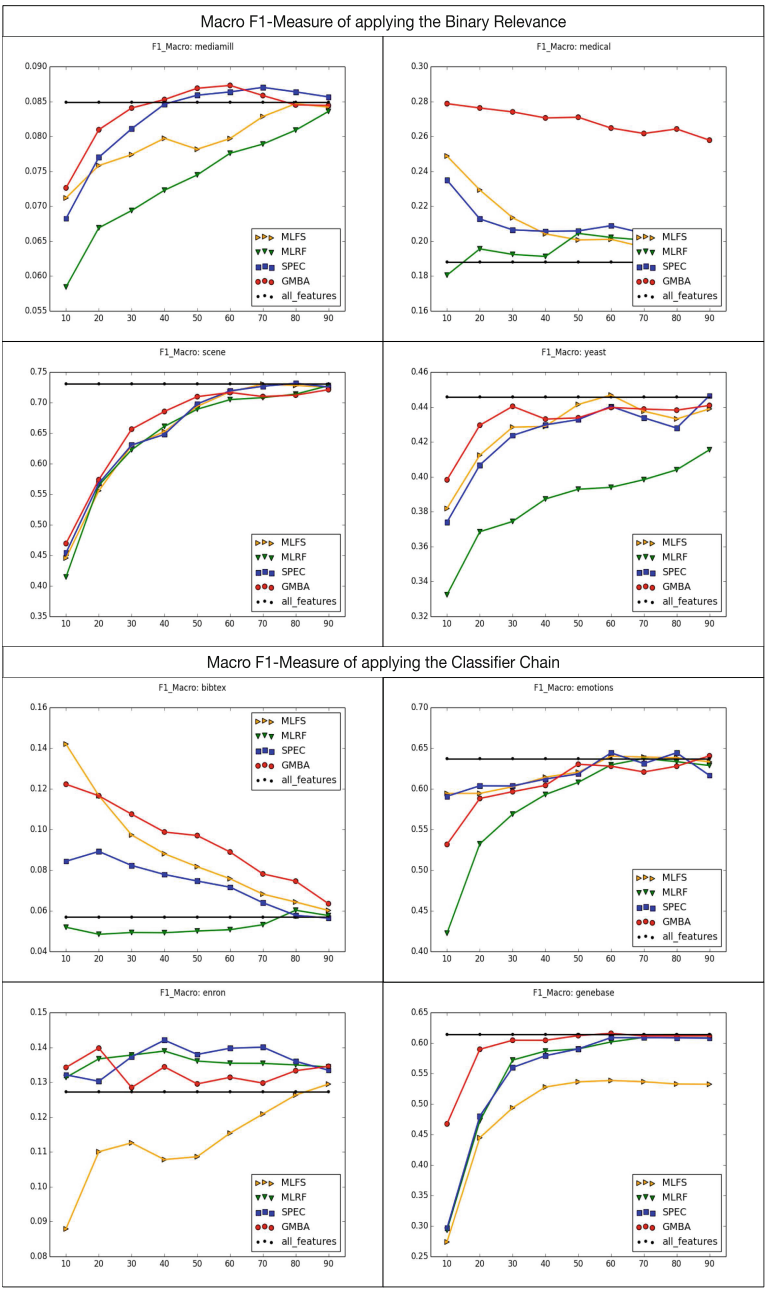

Fig. 4. macro F1-Measure (↑). The first 4 diagrams show the macro F1-Measure of applying the Binary Relevance while the rest show the results from the Classifier Chain. Y-axis corresponds to different metrics and X-axis denotes the percentage of features selected. The horizontal lines are the results of classifying with all features

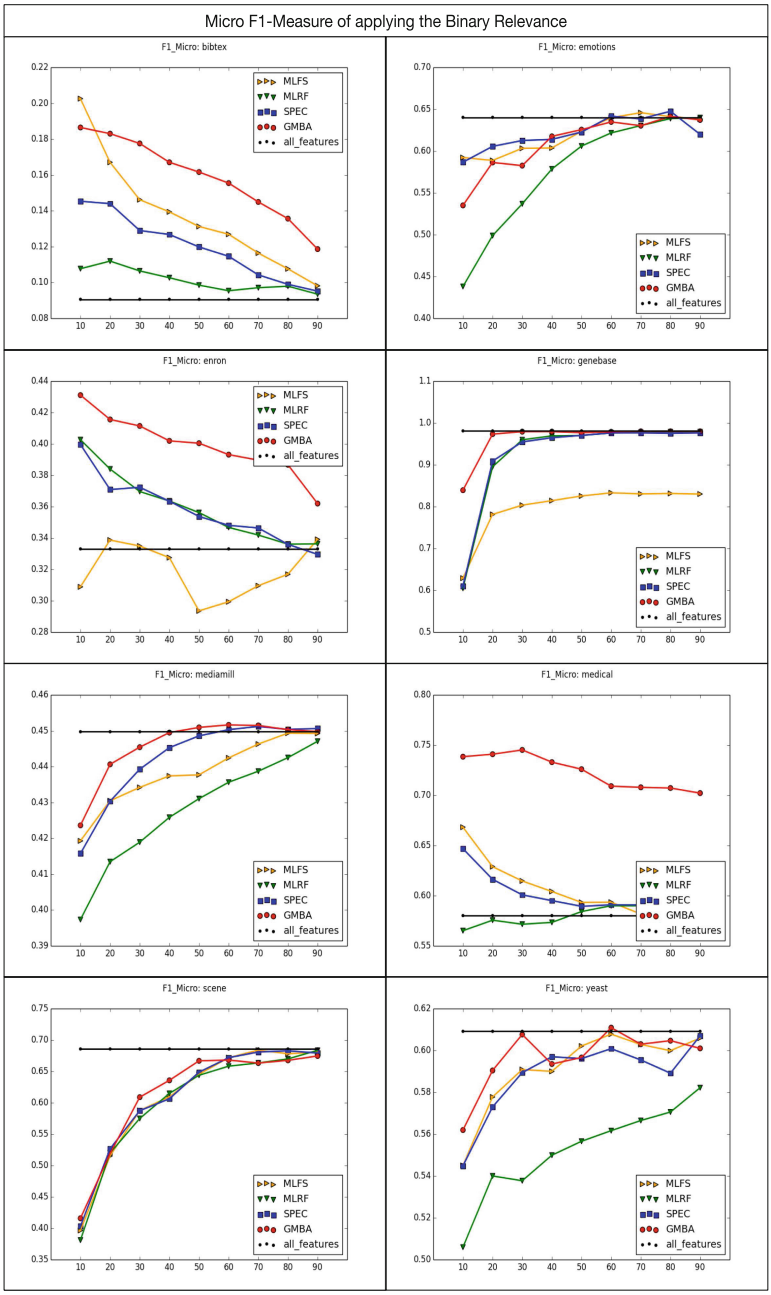

Fig. 5. The micro F1-Measure (↑) of applying the Binary Relevance. Y-axis corresponds to the metrics and X-axis denotes the percentage of features selected. The horizontal lines are the results of classifying with all features

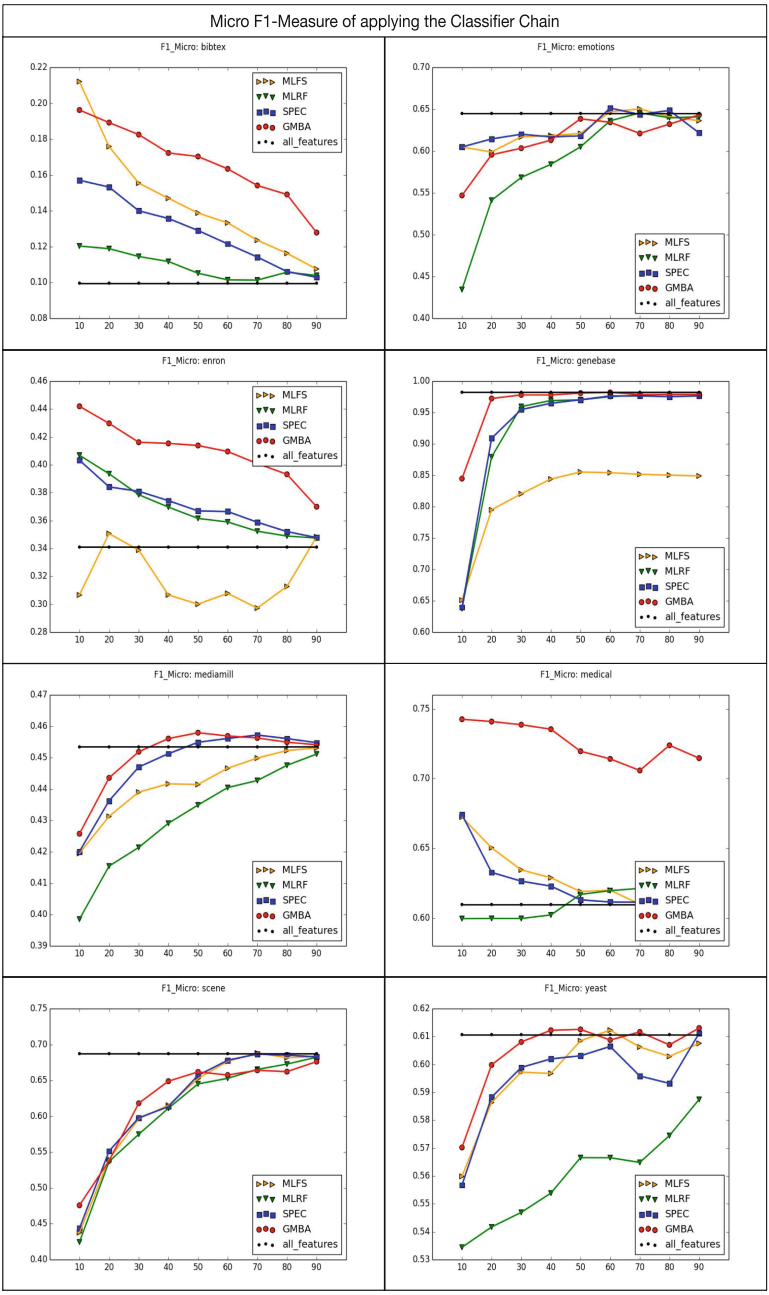

Fig. 6. The micro F1-Measure (↑) of applying the Classifier Chain. Y-axis corresponds to the metrics and X-axis denotes the percentage of features selected. The horizontal lines are the results of classifying with all features

Acknowledgments. This work was partially supported by Natural Science Foundation of Jiangsu Province (BK20131378, BK20140885), National Natural Science Foundation of China (NSFC 41573189, 61300165 and 61300164), Post-doctoral Foundation of Jiangsu Province (1401045C) and sponsored by Qing Lan Project.

References

1. Agarwal, S., Branson, K., Belongie, S.: Higher order learning with graphs. In: International Conference on Machine Learning, pp. 17–24 (2006)
2. Boutell, M.R., Luo, J., Shen, X., Brown, C.M.: Learning multi-label scene classification. Pattern Recogn. **37**(9), 1757–1771 (2004)
3. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph-based image segmentation. Int. J. Comput. Vis. **59**(2), 167–181 (2004)
4. Fürnkranz, J., Hüllermeier, E., Mencía, E.L., Brinker, K.: Multi-label classification via calibrated label ranking. Mach. Learn. **73**(2), 133–153 (2008)
5. Gu, Q., Li, Z., Han, J.: Correlated multi-label feature selection. In: the 20th ACM International Conference on Information and Knowledge Management, pp. 1087–1096 (2011)
6. Huang, H.: Multi-label relieff and f-statistic feature selections for image annotation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2352–2359. IEEE Computer Society, Washington, DC (2012)
7. Huang, S.J., Zhou, Z.H.: Multi-label learning by exploiting label correlations locally. In: AAAI Conference on Artificial Intelligence (2012)
8. Jiang, A., Wang, C., Zhu, Y.: Calibrated rank-svm for multi-label image categorization. In: IEEE International Joint Conference on Neural Networks, pp. 1450–1455 (2008)
9. Lecun, Y., Fu, J.H.: Loss functions for discriminative training of energy-based models. In: The 10th International Workshop on Artificial Intelligence and Statistics (2005)
10. Li, Y., Lu, B.L.: Feature selection based on loss-margin of nearest neighbor classification. Pattern Recogn. **42**(9), 1914–1921 (2009)
11. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. Mach. Learn. **85**(3), 254–269 (2011)
12. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **22**(8), 888–905 (2000)
13. Sun, L., Ji, S., Ye, J.: Hypergraph spectral learning for multi-label classification. In: ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 668–676 (2008)
14. Tang, J., Alelyani, S., Liu, H.: Feature Selection for Classification: A Review. CRC Press, Boca Raton (2012)
15. Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I.P.: Multi-label classification of music into emotions. In: The International Society for Music Information Retrieval (2008)
16. Tsoumakas, G., Katakis, I., Vlahavas, I.: Random k-labelsets for multilabel classification. IEEE Trans. Knowl. Data Eng. **23**(7), 1079–1089 (2010)
17. Tsoumakas, G., Vlahavas, I.: Random k-labelsets: an ensemble method for multilabel classification. In: Kok, J.N., Koronacki, J., Mantaras, R.L., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 406–417. Springer, Heidelberg (2007). doi:10.1007/978-3-540-74958-5_38

18. Wang, S., Siskind, J.M.: Image segmentation with ratio cut. IEEE Trans. Pattern Anal. Mach. Intell. **25**(6), 675–690 (2003)
19. Wang, Y., Li, P., Yao, C.: Hypergraph canonical correlation analysis for multi-label classification. Signal Process. **105**(12), 258–267 (2014)
20. Weinberger, K.Q., Blitzer, J., Saul, L.K.: Distance metric learning for large margin nearest neighbor classification. In: Weiss, Y., Schölkopf, B., Platt, J.C. (eds.) Advances in Neural Information Processing Systems 18, pp. 1473–1480. MIT Press (2006)
21. Zhang, M.L., Peña, J.M., Robles, V.: Feature selection for multi-label naive bayes classification. Inf. Sci. **179**(19), 3218–3229 (2009)
22. Zhang, M.L., Zhang, K.: Multi-label learning by exploiting label dependency. In: Acm SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 999–1008 (2010)
23. Zhang, M.L., Zhou, Z.H.: Ml-knn: A lazy learning approach to multi-label learning. Pattern Recogn. **40**(7), 2038–2048 (2007)
24. Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. **26**(8), 1819–1837 (2014)
25. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: the 24th International Conference on Machine Learning, pp. 1151–1157 (2007)