Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

Towards a unified multi-source-based optimization framework for multi-label learning

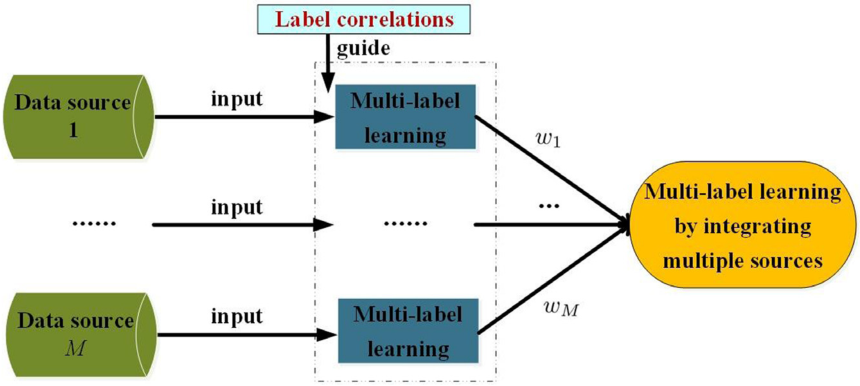
Jia Zhang [a,b], Candong Li [c], Zhenqiang Sun [a,b], Zhiming Luo [d], Changen Zhou [c], Shaozi Li [a,b,*]

[a] Department of Cognitive Science, Xiamen University, Xiamen, 361005, PR China
[b] Fujian Key Laboratory of Brain-inspired Computing Technique and Applications, Xiamen University, Xiamen, 361005, PR China
[c] College of Traditional Chinese Medicine, Fujian University of Traditional Chinese Medicine, Fuzhou, 350122, PR China
[d] Postdoc Center of Information and Communication Engineering, Xiamen University, 361005, PR China

HIGHLIGHTS

- We propose a multi-source-based optimization framework for multi-label learning.
- We consider multi-label consensus learning by preserving the label correlations.
- The proposed method can combine with other multi-label methods freely.
- The proposed method is effective for long-tail data.
- Experiments on various data sets reveal the advantages of the proposed method.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 10 May 2018
Received in revised form 30 November 2018
Accepted 14 December 2018
Available online 20 December 2018

Keywords:
Multi-label learning
Multi-source fusion
Optimization framework

ABSTRACT

In the era of Big Data, a practical yet challenging task is to make learning techniques more universally applicable in dealing with the complex learning problem, such as multi-source multi-label learning. While some of the early work have developed many effective solutions for multi-label classification and multi-source fusion separately, in this paper we learn the two problems together, and propose a novel method for the joint learning of multiple class labels and data sources, in which an optimization framework is constructed to formulate the learning problem, and the result of multi-label classification is induced by the weighted combination of the decisions from multiple sources. The proposed method is responsive in exploiting the label correlations and fusing multi-source data, especially in the fusion of long-tail data. Experiments on various multi-source multi-label data sets reveal the advantages of the proposed method.

1. Introduction

With the development of the information technology from the early 19th century, the expansion size in data is going to be a burning question in practical applications [1,2].

As a example to explain the phenomenon, traditional supervised learning mainly focuses on labeling an unseen instance with single label, and has achieved a great success in reality, such as image recognition [3] and symptom classification in traditional Chinese medicine (TCM) [4]. However, the simplifying assumption, i.e., each instance is relevant to only one class label, is difficult to meet the demand in consideration of that real-world objects might have multiple semantic meanings simultaneously [5,6]. Accordingly, the paradigm of multi-label learning emerges to deal with the learning problem. For multi-label learning, a straight

[*] Corresponding author at: Department of Cognitive Science, Xiamen University, Xiamen, 361005, PR China.

E-mail addresses: zhangjia_gl@163.com (J. Zhang), fjzylcd@126.com (C. Li), m15122329324@163.com (Z. Sun), zhiming.luo@xmu.edu.cn (Z. Luo), 123387260@qq.com (C. Zhou), szlig@xmu.edu.cn (S. Li).

https://doi.org/10.1016/j.asoc.2018.12.016
1568-4946/© 2018 Published by Elsevier B.V.

forward strategy is to transform the learning problem into a set of binary classification subproblems. Nevertheless, the strategy follows the principle that class labels are independent, and has completely ignored the label correlations. As we know, it is essential to improve the learning performance by resolving the correlations among labels in view of that class labels usually have co-occurrence relationships with each other [6,7]. For instance, an image is likely to be annotated with *ship* in case that it is classified as *sea*. To address this problem, many techniques have been adopted to improve the correlation-modeling capability [6], such as information entropy [8], sparse representation [9], and manifold learning [10].

Another important aspect of the expansion size in data is that the growing information comes from multiple and heterogeneous sources with evolving and complex relationships [2,11]. In general, compared with single data source, multiple sources can provide complementary representations for the same object [12–14], which are helpful to the machine learning task, such as multi-label classification. For example, Xie et al. [15] achieved the multi-label consensus learning by introducing the strategy of multi-source learning to counteract the bias of any single data source and the effects of low data quality. Xu et al. [16] made use of graphs to combine different models with image labels, in consideration of the label correlations and the complementary nature of multi-modal features, and proposed a joint learning approach for multi-label image classification. However, the multi-source generation of the aforementioned methods is based on the same feature information using various multi-label classifiers, or descriptors. Moreover, many multi-label ensemble learning approaches have been developed in past years [17,18], whereas they are constructed based on multiple base learners rather than multiple sources. In the literatures [19,20], the multi-source multi-label learning problem is degenerated to the assignment of multiple independent labels, which fails to the label correlations exploitation.

In addition, it is noteworthy that taking all the aspects (multi-label and multi-source) into account has the potential for practical applications. For example, multi-label learning has been employed to achieve drug repositioning considering that a drug has two or more different indications [5], in the meanwhile, the drug has multiple feature representations in terms of its chemical structures, side effects, and target proteins, etc [14,21,22]. Moreover, in TCM diagnosis, a patient is diagnosed with a disease clinically based on a summary of comprehensive signals including inspection, pulse feeling, palpation and the standardized inquiry information [23]. Note that the disease in TCM is denoted by several patterns of syndromes regularly [24–26], therefore, TCM diagnosis can be regarded as a typical multi-source multi-label learning problem.

In this paper, we propose a novel multi-source-based learning approach for multi-label learning, in which multi-source fusion and multi-label prediction can be learned jointly. In detail, we first generate the multi-label prediction based on each data source by preserving the label correlations, and then construct an optimization framework. Under this framework, the assignment of source weights and the weighted combination prediction are iteratively updated to achieve multi-label consensus classification. Finally, extensive experiments reveal the feasibility and effectiveness of the proposed method, and the contributions of this paper are summarized as follow:

- We study a practical yet challenging problem that considers multi-source multi-label classification, and propose a unified optimization framework to the joint learning of multiple labels and data sources.
- Different to the existing multi-label methods, we focus on multi-label consensus learning. Different to the existing methods for multi-source fusion, the proposed method not only can deal with the correlations of multiple labels, but also the special fusion scenario, i.e., long-tail data.
- We propose a novel method to preserve the label correlations, which can be an improved version of some other multi-label learning approaches.
- Extensive experiments on various data sets demonstrate the advantages of the proposed method.

The rest of this paper is organized as follows. Section 2 gives a brief review of multi-label learning and multi-source fusion. Then, we describe the proposed method in detail, and explain the experimental result in Section 4. Finally in Section 5, the conclusion is given.

2. Related work

2.1. Multi-label learning

Comprehensive reviews on multi-label learning can be found in some good surveys [5,6]. Here, we review multi-label methods based on the order of correlations, which can be roughly categorized into three families, as follow: *first-order* strategy is based on the assumption that one binary classifier is designed for each label. By this way, many state-of-the-art methods have been proposed for multi-label learning, and representative algorithms include Binary Support Vector Machine (BSVM) [27] and multi-label learning with Label-specIfic FeaTures (LIFT) [28]. *Second-order* strategy explores pairwise relationships between labels. For achieving the purpose, exploiting the interaction of label pairs is a popular and effective method, which is widely used for the model construction, such as Calibrated Label Ranking (CLR) [29] and the method of Learning Label Specific Features (LLSF) [30], and another method is to optimize the multi-label learning model in which *ranking loss* is involved in the objective function. Based on this, some methods like MultiLabel Consensus Maximization for ranking (MLCM-r) [15] and Mutual Information based multi-label feature selection with Convex Optimization (MICO) [31] have been presented for exploiting the label correlations. In general, compared with *first-order* strategy, *second-order* strategy can achieve better generalization performance with the acceptable complexity. However, the label correlations are possibly detached from the *second-order* limitation in real-world applications. *High-order* strategy tackles multi-label learning problem by mining relations among all labels or subsets of labels, such as Ensembles of Classifier Chains (ECC) [32] and RAndom *k*-labELsets (RA*k*EL) [33]. Apparently *high-order* strategy has the best capability in the label correlations exploitation, whereas the strategy may be less effective due to the time complexity and scalability.

Moreover, many multi-label learning methods have been proposed by considering the label correlations locally. For example, Hou et al. [10] proposed a multi-label manifold method. This method applied locally linear embedding to the reconstruction of label space. Specially, in light of that the label manifold has the consistent local topological structure with the feature manifold, the local correlations can be discovered by learning the label manifold. Huang and Zhou [34] utilized the method of clustering to capture the local label correlations shared by instances, which were incorporated into the multi-label learning via generating a new feature representation.

2.2. Multi-source fusion

As the literature suggested [35,36], multi-source fusion can be mainly divided into two categories: early fusion and late fusion. The strategy of early fusion processes multiple sources in data layer, and the representative method, such as multiple feature concatenation [37], is to concatenate data sources for generating a larger feature representation as input. Huiskes et al. [37] utilized

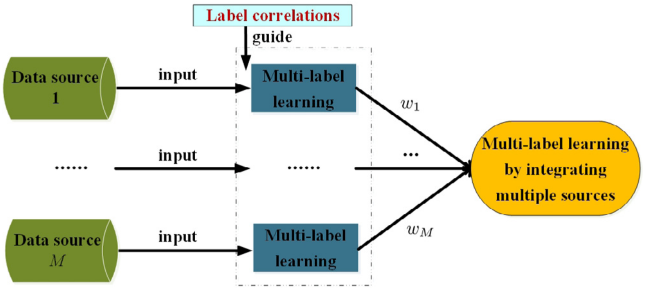


Fig. 1. Paradigm of the proposed method.

the method to SVM classification, and improved the learning performance. Nevertheless, the method ignores the incompatibility of multi-source data. Another strategy, namely late fusion, achieves data fusion in decision-making layer, and many methods have been presented for such data integration [38–41]. Among these methods, Major Voting and Averaging are widely used to the ground truth finding, in which Major Voting puts the highest number of occurrences from multiple sources as the final prediction, while Averaging assumes that all the data sources have equal importance. In light of the privacy and storage of the raw data, these methods have the advantages for the situation, whereas they do not work in the source weight estimation, and thus possibly failing in the fusion of long-tail data [42,43]. In addition, employing graphs to design fusion method has attracted a great attention recently, which has gained promising results for video annotation [44], image clustering [45], and consensus maximization [46]. Unfortunately, these graph-based learning techniques are mainly for designing data graphs based on multi-model features, or multiple models.

Long-tail data is a special type of multi-source data in which most of the sources are *small* sources which make very few claims on all the items, and only a few sources provide much information [43,47]. For example, celebrity information can be found from multiple websites, however, few websites, such as Wikipedia and Baidu, contain a large amount of information about thousands of celebrities, while on many websites, there are only one or several celebrities. To address the challenge, the solution for long-tail data fusion is created from various research fields, such as truth discovery [47] and music recommendation [48]. However, it is not identified in multi-source multi-label learning, and we make the proposed method suitable for this phenomenon.

3. Methodology

In this Section, we describe the proposed framework for multi-label classification by integrating multiple sources, as shown in Fig. 1. Specifically, the classification model is first conducted to generate the prediction based on each data source, in which we resort to each data source with the corresponding prediction to preserve the label correlations for acquiring the optimized result in multi-label setting. After that, in light of the quality of data sources which provide different class-discriminative information, the weights of different data sources are assigned to obtain the final result for multi-label learning.

3.1. The framework

We start by introducing the terms and defining the learning problem. Suppose $\mathcal{D} = \{(\mathbf{x}_i, Y_i)|1 \leq i \leq n\}$ denotes a multi-source multi-label training data set, and $\mathbf{x}_i = \{\mathbf{x}_i^m\}_{m=1}^{M}$ includes all features from M data sources, which is associated with a finite set of q possible class labels $\{l_1, l_2, \ldots, l_q\}$. Let $Y_i = \{y_{i1}, y_{i2}, \ldots, y_{iq}\}$ denotes the set of the ground truth labels of $\mathbf{x}_i$, the corresponding label information of $\mathbf{x}_i$ can be defined: For $y_{ij} \in Y_i$, $y_{ij} = 1$ in case that l_j is relevant to $\mathbf{x}_i$, otherwise y_{ij} to 0.

Based on this, we take the aim at integrating the M data sources for deriving a real-valued function, such that some loss functions or specific evaluation is satisfied, and thus achieving multi-label consensus learning. Considering that the final result for multi-label learning should be close to the predictions from reliable sources, we are inspired to minimize the weighted deviation from the final result to all the predictions (obtained via different data sources), and propose the following optimization objective function, as follow:

$$\min_{\mathcal{W},Y^*} \sum_{m=1}^{M} \frac{w_m}{|C_m|} \sum_{(t,j)\in C_m} \left(Y^*_{tj} - \left(\alpha Y^m_{tj} + (1-\alpha)\, R^m_{tj}\right)\right)^2 \tag{1}$$

$$\text{s.t.} \quad \delta(\mathcal{W}) = 1,\ w_m > 0,$$

In this optimization problem, the distribution of source weights is represented by $\mathcal{W} = \{w_1, w_2, \ldots, w_M\}$ where w_m denotes the weight of the mth data source, and $\delta(\mathcal{W})$ is the regularization function. $Y^*_{tj} \in Y^*$ is the final prediction result of unseen instance $\mathbf{x}_t$ on label l_j, both of Y^m_{tj} and R^m_{tj} are the predictions from the mth data source, and $\alpha \in [0, 1]$ is the parameter trading off the two values. Note that not all instances can predict real values to all associated labels, hence the item $\frac{1}{|C_m|}$ is added to minimize the average deviation, in which C_m is the collection of (t, j) which meets the condition $Y^m_{tj} + R^m_{tj} > 0$.

By minimizing the objective function, we try to obtain the final result Y^* for multi-label learning by incorporating all data sources into the unified optimization framework. By this way, Y^* is closer to the predictions from data sources with larger weights, and vice versa.

3.2. Multi-label prediction from single source

As shown in Eq. (1), for predicting the final result Y^* of test data on multiple class labels, we should generate multi-label prediction based on each data source in advance. Thus, two steps are taken to make multi-label prediction from single data source. First, a classification model is applied to multi-label learning in a label-by-label style. Suppose Y^m is the multi-label classification result based on the mth data source, arbitrary element $Y^m_{tj} \in Y^m$ can be obtained as follow:

$$Y^m_{tj} = \{l_j | g_j(\mathbf{x}^m_t) > 0\}, \tag{2}$$

where g_j is the classification model for label l_j. In fact, each instance possibly attaches to q labels, hence a family of q classification models $\{g_1, g_2, \ldots, g_q\}$ are in preparation for producing its associated label set. Considering that the *first-order* strategy which can provide good performance on *hamming loss* [30], we simply implement the LIBrary for Support Vector Machines (LIBSVM) (with linear kernel) [49] to the above binary classifier induction for the sake of the label correlations exploitation favorably.

Based on each data source $\mathbf{x}^m$ with its corresponding prediction Y^m, the correlations between labels are further explored. In light of that instances close to each other are more likely to share a label [10], we capture the label correlations by the similarity computing which employs both feature space and label space to search for similar neighbors of test data, as follow:

$$sim^m(t, z) = \beta sim^m_{fea}(t, z) + (1 - \beta) sim^m_{lab}(t, z), \tag{3}$$

In Eq. (3), similarity score $sim^m(t, z)$ is derived by summing up the weighted similarities. In which $sim^m_{fea}(t, z)$ denotes the similarity between two instances $\mathbf{x}^m_t$ and $\mathbf{x}^m_z$, and the correlation is calculated by the Cosine similarity. Similarly, $sim^m_{lab}(t, z)$ denotes the similarity calculated by the Jaccard coefficient according to the label information, and β is the parameter ranging from 0 to 1. Then, the prediction of unseen instance $\mathbf{x}^m_t$ on label l_j can be recovered by employing its Top-k nearest neighbors, as follow:

$$R^m_{tj} = \frac{\sum_{z\in Z} sim^m(t,z) \times Y_{zj}}{\sum_{z\in Z} sim^m(t,z)}, \tag{4}$$

where Y_{zj} is the ground truth of $\mathbf{x}^m_z$ on label l_j, Z denotes the neighbor set of $\mathbf{x}^m_t$, and $|Z| = k$. Accordingly, the information of the label correlations based on the mth data source can be preserved in character matrix R^m.

Algorithm 1 Multi-label learning by preserving the label correlations (MLLC)

Input: Multi-label training data set $\mathcal{D} = \{(\mathbf{x}^m_i, Y_i)|1 \le i \le n\}$, parameters α, β, and k.
Output: The predicted label set for unseen instance $\mathbf{x}^m_t$.

```
Employ SVM classifier to generate $Y^m_t$;
for $i = 1$ to $n$ do
    Compute similarity matrix $sim^m(t, i)$ using Eq. (3);
end for
Generate neighbor set $Z$ of $\mathbf{x}^m_t$ according to $\{sim^m(t, i)^n_{i=1}\}$;
Predict character matrix $R^m_t$ using Eq. (4);
Obtain $\mathbf{x}^m_t$'s label set by computing $\alpha Y^m_t + (1-\alpha) R^m_t$.
```

3.3. Solution for multi-label consensus learning

Multi-label prediction can be made from single data source, and the pseudo code is shown in Algorithm 1. Based on this, we consider to integrate multiple sources by dealing with the optimization problem Eq. (1), and use the iterative strategy to develop a practical method which is applicable to the source weight estimation, thus achieving multi-label consensus learning.

3.3.1. Solving $\mathcal{W}$

We derive the distribution of source weights $\mathcal{W}$ by fixing Y^*. Let $w_m = s^2_m$, and the regularization function is defined as $\delta(\mathcal{W}) = \sum_{m=1}^M s_m$, and thus the optimization problem with respect to $\mathcal{W}$ is transformed into a constrained quadratic programming problem.

$$\min_{\mathcal{W}} \sum_{m=1}^{M} \frac{s^2_m}{|C_m|} e^m_{dev} \quad \text{s.t.} \quad \sum_{m=1}^{M} s_m = 1, s_m > 0, \tag{5}$$

where $e^m_{dev} = \sum_{(t,j)\in C_m} \left(Y^*_{tj} - \left(\alpha Y^m_{tj} + (1-\alpha) R^m_{tj}\right)\right)^2$. To solve this problem, one way to circumvent this difficulty leading to a meaningful weight update is the method of Lagrange multipliers, and we define the Lagrangian of Eq. (5):

$$L(\mathcal{W}, \lambda) = \sum_{m=1}^{M} \frac{s^2_m}{|C_m|} e^m_{dev} + \lambda(\sum_{m=1}^{M} s_m - 1), \tag{6}$$

Considering the partial derivatives of the Lagrangian as zero in terms of the Lagrange multiplier λ and s_m, we can infer the following formulas:

$$\begin{cases} \dfrac{\partial L(\mathcal{W},\lambda)}{\partial s_m} = 2\dfrac{s_m}{|C_m|} e^m_{dev} + \lambda = 0, \quad (m = 1, 2, \ldots, M) \\ \dfrac{\partial L(\mathcal{W},\lambda)}{\partial \lambda} = \displaystyle\sum_{m=1}^{M} s_m - 1 = 0. \end{cases} \tag{7}$$

In Eq. (7), $M + 1$ equations are established involving $M + 1$ independent variables, namely $\{s_m\}^M_{m=1}$ and λ, consequently, the optimal solution of $\mathcal{W}$ can be calculated as follow:

$$s_m = \frac{|C_m|}{e^m_{dev}} \left(\sum_{m'=1}^{M} \frac{|C_{m'}|}{e^{m'}_{dev}}\right)^{-1}, \tag{8}$$

In Eq. (8), the weight of the mth data source is inversely proportional to the average deviation of the prediction in terms of Y^m and R^m, and the smaller value of $\frac{e^m_{dev}}{|C_m|}$ indicates the larger weight of the mth data source, and vice versa.

3.3.2. Solving Y^*

In this step, the distribution of source weights $\mathcal{W}$ is fixed, and we solve for multi-label classification result Y^* by taking the weighted predictions into account. Since each value $Y^*_{tj} \in Y^*$ can be updated independently, the original optimization problem Eq. (1) becomes the following one:

$$\min_{Y^*_{tj}} \sum_{m=1}^{M} \frac{s^2_m}{|C_m|} \left(Y^*_{tj} - \left(\alpha Y^m_{tj} + (1-\alpha) R^m_{tj}\right)\right)^2 \quad \text{s.t.} \quad Y^m_{tj} + R^m_{tj} > 0, \tag{9}$$

Setting the gradient of Eq. (9) w.r.t. Y^*_{tj} to zero, the optimal Y^*_{tj} corresponding to label l_j with respect to instance $\mathbf{x}_t$ can be solved, as follow

$$Y^*_{tj} = \frac{\sum_{m=1}^{M} \frac{s^2_m}{|C_m|} \left(\alpha Y^m_{tj} + (1-\alpha) R^m_{tj}\right)}{\sum_{m=1}^{M} \frac{s^2_m}{|C_m|}}, \tag{10}$$

As shown in Eq. (10), Y^*_{tj} is closely relevant to the source weights and all the predictions according to different data sources, which tends to be assigned a value based on the predictions from reliable data sources. In order to provide a clear description, we display the proposed algorithm in Algorithm 2.

Algorithm 2 Multi-source-based optimization framework for multi-label classification (MLSO)

Input: Multi-source multi-label training data set $\mathcal{D} = \{(\mathbf{x}_i, Y_i)|1 \le i \le n\}$, parameters α, β, and k.
Output: The predicted label set Y^*_t for unseen instance $\mathbf{x}_t$.

```
Initialize the distribution of source weights $\mathcal{W} = \{w_1, w_2, \ldots, w_M\}$;
for $m = 1$ to $M$ do
    Employ SVM classifier to generate $Y^m_t$;
    for $i = 1$ to $n$ do
        Compute similarity matrix $sim^m(t, i)$ using Eq. (3);
    end for
    Generate neighbor set $Z$ of $\mathbf{x}^m_t$ according to $\{sim^m(t, i)^n_{i=1}\}$;
    Predict character matrix $R^m_t$ using Eq. (4);
end for
repeat
    for $j = 1$ to $q$ do
        if $\sum_{m=1}^M (Y^m_{tj} + R^m_{tj}) = 0$ then
            $Y^*_{tj} = 0$;
        else
            Update $Y^*_{tj}$ using Eq. (10);
        end if
    end for
    Compute $\{s_m\}^M_{m=1}$ using Eq. (8) to update $\mathcal{W}$;
until Convergence;
return The predicted label set $Y^*_t$.
```

3.4. Practical issues

Several critical issues are discussed to make the proposed method practical and complete. First, the designed objective function is an optimization problem involving two unknown variable

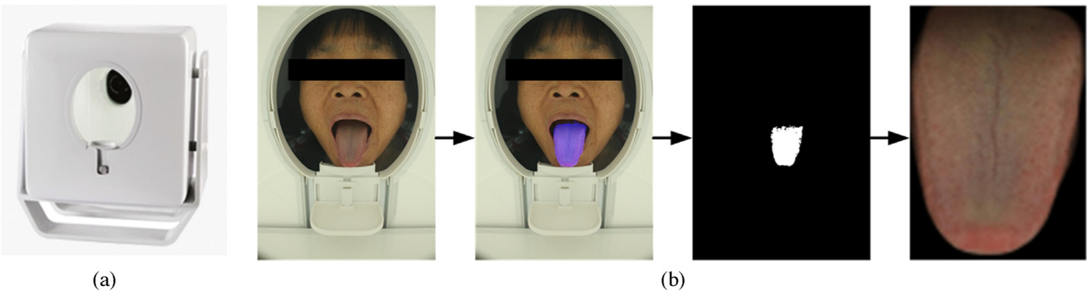


Fig. 2. (a) The tongue capture device consists of a video camera placed in the center and fluorescent tubes situated symmetrically on either sides of the camera in order to produce a uniform illumination. (b) Tongue image segmentation is achieved by the U-Net neural network [50] for automatic feature extraction based on a cleaner tongue color.

sets, i.e., $\mathcal{W}$ and Y^*, nevertheless, the objective function is not convex in consideration of that it is difficult to recover both unknowns in the meanwhile. Thus, we introduce an alternating optimization for the proposed method. By setting $w_m = s_m^2$, and $\delta(\mathcal{W}) = \sum_{m=1}^{M} s_m$, we first prove that the optimization problem w.r.t. $\mathcal{W}$, as shown in Eq. (5), is a convex program. In addition, the minimization problem w.r.t. Y_{ij}^* defined in Eq. (9), is also a convex problem which can be easily solved by gradient-based optimization. As the literature suggests [51] that any local minimum is also a global minimum for a convex problem, hence we can conclude that the solution of the proposed method converges to the global minimum.

Some methods, i.e. Major Voting and Averaging, are effective ways to the initialization of iterative optimization, whereas the methods may be not suitable for the real scenario with long-tail phenomenon, which easily cause the *small* source noticed with a larger weight. To solve the problem, for estimating the weight of a data source, we should not only pay close attention to the prediction information provided by this source, but also the number of the source's positive predictions. Explanatorily, the predicted value is not less than $\frac{1}{2}$. Thus, we initialize each source weight using the proportion of the positive predictions from the total number of positive predictions.

$$s_m = \frac{|\{(t,j)|\alpha Y_{tj}^m + (1-\alpha)\, R_{tj}^m \geq \frac{1}{2}\}|}{\sum_{m=1}^{M} |\{(t,j)|\alpha Y_{tj}^m + (1-\alpha)\, R_{tj}^m \geq \frac{1}{2}\}|}, \tag{11}$$

As each data source possesses the same predictive power in terms of the number of positive predictions, the initialization of the proposed optimization framework is set as $\{s_m\}_{m=1}^{M} = \frac{1}{M}$.

Finally, we analyze the complexity of our method, which mainly includes two parts: the label correlations exploitation and multi-source fusion. For exploiting the label correlations, the time cost is dominated by the similarity computing, and leads to a complexity of $O(np)$, where n is the number of instances for training, and p is the number of instances for test. For multi-source fusion, the total complexity is $O(\sum_{m=1}^{M} |C_m|)$, where M is the number of data sources, and $|C_m|$ is the number of nonzero predictions based on data source m. Due to $\{|C_m|\}_{m=1}^{M} \leq pq$, we can conclude that the time cost for consensus-based multi-label classification is less than $O(Mpq)$, where q is the number of labels.

4. Experiments

In this Section, extensive experiments are conducted to validate the proposed method. First, we describe the used data sets and experimental setup, and then verify the effectiveness of the proposed method. Finally, the parameter sensitivity and convergence are analyzed.

4.1. Data sets

`Drug repositioning (DR)`: 536 approved drugs are collected from DrugBank [52], and chemical structures, side effects, and target proteins of the 536 approved drugs are also collected as three data sources to identify 578 kinds of diseases. Particularly, the drug–disease associations are extracted from the National Drug File-Reference Terminology, which is one of parts of Unified Medical Language System (UMLS) [53]. In addition, chemical structures of each drug are obtained from a Public repository for Chemical structures (PubChem) [54]. Each drug is represented by an 881-dimensional chemical substructure, and the presence or absence of each substructure is denoted by 1 or 0, respectively. Moreover, side effects of the 536 approved drugs are collected from the SIDER database [55], which results in a collection of relationships between these drugs and 1252 side-effects keywords. Finally, we extract drug–target pairs from DrugBank [52], and construct the data source in which 563 target proteins are referred to the 536 drugs.

`Syndrome diagnosis in TCM`: The information of 729 patients is collected from the Second People's Hospital Health Management of Fujian Province, which is for the purpose of the multi-label learning task about syndrome diagnosis in TCM. Concretely, the TCM data set is created with 421 symptoms and 339 associated class labels, each of which indicates a syndrome of patients. In addition, tongue images of the 729 patients are also captured by professional tongue capture device, as shown in Fig. 2(a). Specifically, a patient first places his or her chin on a chin rest, and then fully exposes the tongue to obtain corresponding images by changing the position or height of the chin rest. Taking the quality of images into consideration, we save all the images in JPEG format with 3168 × 4752 size, which are color corrected to eliminate the variability caused by the device dependence and changes of the illumination. Then we further process original tongue images into the images which only contain the message of the tongue, and the preprocessing procedure is shown in Fig. 2(b). Finally, we describe the images by 256-dimensional HSV color space histogram, 768-dimensional LAB color space histogram, 768-dimensional RGB color space histogram respectively, which can be viewed as the other data sources for syndrome diagnosis.

Moreover, we use Corel5k data set to evaluate algorithms on the task of automatic image annotation, which is public available and can be downloaded from the website (http://lear.inrialpes.fr/people/guillaumin/data.php). This data set contains 4999 images collected from the larger Corel CD set, each of which is manually annotated with keywords from a dictionary of 260 distinct terms. For automatic image annotation, we use four different visual descriptors to extract features, which are DenseSift, Gist, HarrisHue, and Rgb [56].

The detailed data description of the three data sets is demonstrated in Table 1. DR data set contains three data sources, and

Table 1
The description of the experimental data sets.

Data set	Data source	#Instances	#Features	#Labels	Cardinality
DR	CHST	536	881	578	4.1586
	SIEF		1252		
	PRTA		563		
TCM	SYMP	729	421	339	6.0247
	THSV		256		
	TLAB		768		
	TRGB		768		
Corel5k	DESI	4999	1000	260	3.3965
	GIST		512		
	HAHU		100		
	RGB		4096		

the data sources of chemical structures, side effects, and target proteins are represented by CHST, SIEF, and PRTA respectively. TCM data set contains four data sources, in which SYMP denotes the data source including a total of 421 symptoms, and THSV, TLAB, and TRGB represent three data sources formed by extracted features from HSV, LAB, and RGB color spaces based on all the tongue images respectively. Corel5k data sets contains four data sources, i.e., DESI, GIST, HAHU, and RGB, which are formed by extracted features using DenseSift, Gist, HarrisHue, and Rgb descriptors respectively. In addition, the cardinality measures the average number of relevant labels of each instance.

4.2. Experimental setup

In experiments, 10-fold cross validation is applied to evaluating the learning performance systematically. As an explanation, all samples are randomly divided into 10 equal subsets, each subset is held-out in turn for test, while the remaining data is merged to training. As the validation is iterated 10 times, the averaged metric values out of ten runs are calculated for the algorithms. Next, we introduce the evaluation metrics and the comparing algorithms.

4.2.1. Evaluation metrics

We employ five widely used multi-label evaluation metrics to measure the learning performance [6,57]. Given test set $T' = \{(\mathbf{x}_i, Y_i)|1 \leq i \leq t\}$ and the family of q learned functions $\{f_1, f_2, \ldots, f_q\}$, we can predict a relevant label set Y_i' for unseen instance $\mathbf{x}_i$.

Hamming loss: The metric evaluates the fraction of instance-label pairs which have been misclassified. Suppose Δ corresponds to the symmetric difference between the two sets.

$$\mathtt{hloss} = \frac{1}{tq}\sum_{i=1}^{t}|Y_i'\Delta Y_i|, \tag{12}$$

One-error: The metric evaluates the fraction of examples whose top-ranked label is not in the relevant label set.

$$\mathtt{one\text{-}error} = \frac{1}{t}\sum_{i=1}^{t}[\![[\arg\max_{l_k \in L} f_k(\mathbf{x}_i)] \notin Y_i]\!], \tag{13}$$

Coverage: The metric evaluates how many steps are needed, on average, to go down the label ranking list so as to cover all the ground-truth labels of the instance. Suppose $rank(\mathbf{x}_i, l_k) = \sum_{j=1}^{q}[\![f_j(\mathbf{x}_i) \geq f_k(\mathbf{x}_i)]\!]$ returns the rank of l_k when all labels in L are sorted in descending order based on the q functions.

$$\mathtt{coverage} = \frac{1}{q}\left(\frac{1}{t}\sum_{i=1}^{t}\max_{l_k \in Y_i} rank(\mathbf{x}_i, l_k) - 1\right), \tag{14}$$

Ranking loss: The metric evaluates the fraction of reversely ordered label pairs. Suppose $\bar{Y}_i$ is the complementary set of Y_i in L.

$$\mathtt{rloss} = \frac{1}{t}\sum_{i=1}^{t}\frac{|\{(l_j, l_k)|f_j(\mathbf{x}_i) \leq f_k(\mathbf{x}_i), (l_j, l_k) \in Y_i \times \bar{Y}_i\}|}{|Y_i||\bar{Y}_i|}, \tag{15}$$

Average precision: The metric evaluates the average fraction of relevant labels ranked higher than a particular label $l_k \in Y_i$.

$$\mathtt{avgprec} = \frac{1}{t}\sum_{i=1}^{t}\frac{1}{|Y_i|}\sum_{l_j, l_k \in Y_i}\frac{|\{l_j|rank(\mathbf{x}_i, l_j) \leq rank(\mathbf{x}_i, l_k)\}|}{rank(\mathbf{x}_i, l_k)}, \tag{16}$$

In brief, the five metrics evaluate the algorithm performance from various aspects. For *hamming loss*, *one-error*, *coverage* and *ranking loss*, the smaller the values the better the performance. For *average precision*, the larger the value the better the performance.

4.2.2. Comparing algorithms

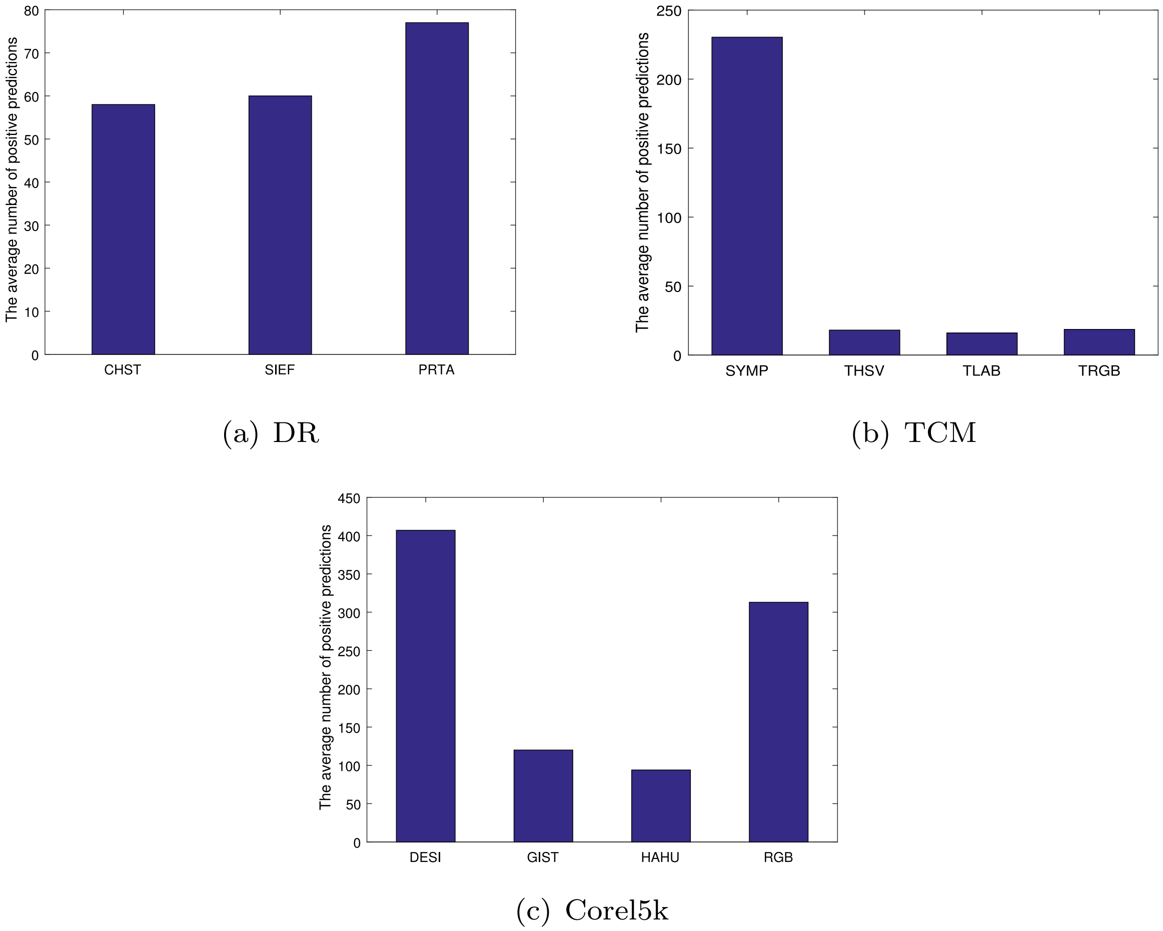
For verifying the effectiveness of our proposed algorithm in the label correlations exploitation. MLLC is first compared with BSVM (with linear kernel) [27], which learns a binary SVM for each label, and can be regarded as a degenerated version of MLLC. Furthermore, in light of that LIFT [28] is helpless in the label correlations exploitation, we utilize LIFT as the multi-label classifier instead of BSVM to verify whether the proposed method can be regarded as

Table 2
Comparative results (mean±std. Deviation) with BSVM as the multi-label classifier.

Data source	Method	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
CHST	MLLC	**0.0067**±0.0006	0.6757 ± 0.0815	**0.4298**±0.0646	**0.2386**±0.0387	**0.3030**±0.0572
	BSVM	0.0069 ± 0.0008	**0.6713**±0.0622	0.4537 ± 0.0626	0.2574 ± 0.0362	0.2844 ± 0.0555
SIEF	MLLC	**0.0064**±0.0007	**0.5450**±0.0618	**0.3802**±0.0591	**0.2049**±0.0343	**0.4020**±0.0486
	BSVM	0.0065 ± 0.0008	0.5728 ± 0.0418	0.4126 ± 0.0555	0.2182 ± 0.0330	0.3646 ± 0.0401
PRTA	MLLC	**0.0068**±0.0009	**0.5242**±0.0332	**0.3993**±0.0630	**0.2167**±0.0369	**0.4158**±0.0451
	BSVM	**0.0068**±0.0009	0.5916 ± 0.0469	0.4854 ± 0.0621	0.2819 ± 0.0471	0.3253 ± 0.0440
SYMP	MLLC	**0.0122**±0.0008	0.1701 ± 0.0330	**0.3251**±0.0435	**0.0617**±0.0081	**0.6405**±0.0265
	BSVM	0.0123 ± 0.0009	**0.1358**±0.0263	0.3453 ± 0.0436	0.0665 ± 0.0076	0.6355 ± 0.0177
DESI	MLLC	**0.0118**±0.0002	0.4611 ± 0.0187	**0.1851**±0.0082	**0.0795**±0.0049	**0.4561**±0.0094
	BSVM	**0.0118**±0.0002	**0.4549**±0.0154	0.1995 ± 0.0094	0.0829 ± 0.0047	0.4469 ± 0.0083
GIST	MLLC	**0.0127**±0.0001	**0.6147**±0.0201	**0.2568**±0.0070	**0.1138**±0.0029	**0.3457**±0.0085
	BSVM	0.0128 ± 0.0001	0.6243 ± 0.0209	0.2713 ± 0.0067	0.1184 ± 0.0035	0.3290 ± 0.0079
HAHU	MLLC	**0.0127**±0.0001	**0.6451**±0.0284	**0.2656**±0.0091	**0.1178**±0.0039	**0.3217**±0.0101
	BSVM	0.0128 ± 0.0001	0.6859 ± 0.0154	0.3082 ± 0.0094	0.1384 ± 0.0040	0.2724 ± 0.0089
RGB	MLLC	**0.0123**±0.0002	**0.5375**±0.0258	**0.2229**±0.0094	**0.0964**±0.0039	**0.4052**±0.0148
	BSVM	0.0124 ± 0.0002	0.5717 ± 0.0123	0.2494 ± 0.0123	0.1066 ± 0.0047	0.3680 ± 0.0113

Table 3
Comparative results (mean±std. Deviation) with LIFT as the multi-label classifier.

Data source	Method	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
CHST	MLLC	**0.0067**±0.0005	**0.7018**±0.0817	**0.4361**±0.0340	**0.2526**±0.0313	**0.2727**±0.0550
	LIFT	**0.0067**±0.0005	0.7219 ± 0.0709	0.4682 ± 0.0367	0.2743 ± 0.0308	0.2343 ± 0.0413
SIEF	MLLC	0.0064 ± 0.0005	**0.5942**±0.0601	**0.4185**±0.0411	**0.2307**±0.0307	**0.3682**±0.0489
	LIFT	**0.0063**±0.0004	0.5982 ± 0.0817	0.4587 ± 0.0484	0.2607 ± 0.0355	0.3428 ± 0.0587
PRTA	MLLC	**0.0070**±0.0006	**0.5687**±0.0567	**0.4240**±0.0502	**0.2351**±0.0346	**0.4016**±0.0459
	LIFT	0.0074 ± 0.0013	0.6901 ± 0.1078	0.4686 ± 0.0491	0.2703 ± 0.0374	0.2715 ± 0.0635
SYMP	MLLC	**0.0133**±0.0009	0.2333 ± 0.0396	**0.3911**±0.0429	**0.0802**±0.0092	**0.5942**±0.0272
	LIFT	**0.0133**±0.0009	**0.1990**±0.0377	0.4168 ± 0.0408	0.0882 ± 0.0085	0.5910 ± 0.0260
DESI	MLLC	**0.0121**±0.0002	**0.4785**±0.0159	**0.2036**±0.0137	**0.0874**±0.0060	**0.4411**±0.0119
	LIFT	**0.0121**±0.0002	0.4797 ± 0.0200	0.2445 ± 0.0141	0.1026 ± 0.0056	0.4223 ± 0.0105
GIST	MLLC	**0.0126**±0.0002	**0.5955**±0.0254	**0.2615**±0.0074	**0.1159**±0.0030	**0.3507**±0.0131
	LIFT	0.0127 ± 0.0001	0.6037 ± 0.0237	0.2861 ± 0.0069	0.1250 ± 0.0033	0.3299 ± 0.0125
HAHU	MLLC	**0.0126**±0.0002	**0.6395**±0.0268	**0.2766**±0.0103	**0.1227**±0.0036	**0.3241**±0.0090
	LIFT	0.0127 ± 0.0002	0.6531 ± 0.0316	0.3483 ± 0.0143	0.1562 ± 0.0058	0.2833 ± 0.0128
RGB	MLLC	**0.0125**±0.0002	**0.5535**±0.0191	**0.2434**±0.0105	**0.1037**±0.0052	**0.3933**±0.0093
	LIFT	**0.0125**±0.0002	0.5773 ± 0.0200	0.2948 ± 0.0099	0.1253 ± 0.0057	0.3598 ± 0.0103



(a) DR

(b) TCM

(c) Corel5k

Fig. 3. The average number of positive predictions according to different data sources.

an improved version of LIFT. In addition, to highlight the superiority in multi-source fusion, we conduct the experiment for data source comparison, and also compare the proposed algorithm with two late fusion methods, i.e., Major Voting and Averaging, as the literature [15] suggested.

For all the comparing algorithms, the parameter of each algorithm is set to the default setting. For the proposed method, parameters α and β are searched in $\{0.1, 0.2, \ldots, 1\}$, and k is searched in $\{10, 20, \ldots, 100\}$.

4.3. Multi-label learning

In this Section, we compare the proposed method with other methods in multi-label learning. Tables 2–3 report the averaged performance regarding each evaluation metric, in which the experimental result is generated based on single data source from DR, TCM, and Corel5k data sets, and the better performance between two algorithms is highlighted in boldface. According to Tables 2–3, the experimental result is analyzed as follow:

Table 4
Data source comparison (mean±std. Deviation) on DR data set.

Data source	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
CHST	0.0067 ± 0.0006	0.6757 ± 0.0815	0.4298 ± 0.0646	0.2386 ± 0.0387	0.3030 ± 0.0572
SIEF	0.0064 ± 0.0007	0.5450 ± 0.0618	0.3802 ± 0.0591	0.2049 ± 0.0343	0.4020 ± 0.0486
PRTA	0.0068 ± 0.0009	0.5242 ± 0.0332	0.3993 ± 0.0630	0.2167 ± 0.0369	0.4158 ± 0.0451
All sources	**0.0062**±0.0008	**0.4927**±0.0513	**0.3575**±0.0648	**0.1853**±0.0333	**0.4550**±0.0511

Table 5
Data source comparison (mean±std. Deviation) on TCM data set.

Data source	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
SYMP	0.0122 ± 0.0010	0.1701 ± 0.0330	**0.3251**±0.0435	**0.0617**±0.0081	0.6405 ± 0.0265
THSV	0.0180 ± 0.0014	0.5624 ± 0.0657	0.4090 ± 0.0443	0.0978 ± 0.0071	0.3536 ± 0.0203
TLAB	0.0181 ± 0.0015	0.6090 ± 0.0470	0.4122 ± 0.0412	0.0982 ± 0.0058	0.3447 ± 0.0160
TRGB	0.0181 ± 0.0015	0.5968 ± 0.0595	0.4075 ± 0.0413	0.0968 ± 0.0083	0.3516 ± 0.0241
All sources	**0.0118**±0.0011	**0.1604**±0.0272	0.3328 ± 0.0634	0.0637 ± 0.0108	**0.6473**±0.0191

Table 6
Data source comparison (mean±std. Deviation) on Corel5k data set.

Data source	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
DESI	0.0119 ± 0.0002	0.4611 ± 0.0187	0.1851 ± 0.0082	0.0795 ± 0.0049	0.4561 ± 0.0094
GIST	0.0127 ± 0.0001	0.6147 ± 0.0201	0.2568 ± 0.0070	0.1138 ± 0.0029	0.3457 ± 0.0085
HAHU	0.0127 ± 0.0001	0.6451 ± 0.0284	0.2656 ± 0.0091	0.1178 ± 0.0039	0.3217 ± 0.0101
RGB	0.0123 ± 0.0002	0.5375 ± 0.0258	0.2229 ± 0.0094	0.0964 ± 0.0039	0.4052 ± 0.0148
All sources	**0.0117**±0.0002	**0.4349**±0.0146	**0.1716**±0.0079	**0.0708**±0.0037	**0.4840**±0.0074

Table 7
Method comparison (mean±std. Deviation) on DR data set.

Method	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
MLSO	**0.0062**±0.0008	**0.4927**±0.0513	**0.3575**±0.0648	**0.1853**±0.0333	**0.4550**±0.0511
MLLC-A	**0.0062**±0.0007	0.4947 ± 0.0570	0.3584 ± 0.0652	0.1860 ± 0.0343	0.4535 ± 0.0488
BSVM-A	0.0064 ± 0.0008	0.5231 ± 0.0544	0.3919 ± 0.0610	0.2100 ± 0.0353	0.4039 ± 0.0512
MLLC-V	0.0063 ± 0.0007	0.5654 ± 0.0685	0.3640 ± 0.0609	0.1903 ± 0.0344	0.4034 ± 0.0566
BSVM-V	0.0065 ± 0.0008	0.5784 ± 0.0566	0.4006 ± 0.0551	0.2164 ± 0.0339	0.3590 ± 0.0473

Table 8
Method comparison (mean±std. Deviation) on TCM data set.

Method	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
MLSO	**0.0118**±0.0011	**0.1604**±0.0272	**0.3328**±0.0634	**0.0637**±0.0108	**0.6473**±0.0191
MLLC-A	0.0164 ± 0.0019	0.2236 ± 0.0462	0.3575 ± 0.0579	0.0744 ± 0.0106	0.5146 ± 0.0225
BSVM-A	0.0161 ± 0.0019	0.2386 ± 0.0350	0.3716 ± 0.0578	0.0778 ± 0.0102	0.5222 ± 0.0190
MLLC-V	0.0177 ± 0.0020	0.4156 ± 0.0662	0.3581 ± 0.0577	0.0766 ± 0.0103	0.4382 ± 0.0270
BSVM-V	0.0177 ± 0.0020	0.4073 ± 0.0621	0.3715 ± 0.0578	0.0803 ± 0.0101	0.4422 ± 0.0242

Table 9
Method comparison (mean±std. Deviation) on Corel5k data set.

Method	*Hamming loss*↓	*One-error*↓	*Coverage*↓	*Ranking loss*↓	*Average precision*↑
MLSO	**0.0117**±0.0002	0.4349 ± 0.0146	**0.1716**±0.0079	**0.0708**±0.0037	**0.4840**±0.0074
MLLC-A	0.0123 ± 0.0001	**0.4287**±0.0200	0.1831 ± 0.0089	0.0748 ± 0.0037	0.4724 ± 0.0088
BSVM-A	0.0125 ± 0.0001	0.4289 ± 0.0134	0.2120 ± 0.0088	0.0861 ± 0.0032	0.4514 ± 0.0084
MLLC-V	0.0120 ± 0.0001	0.4717 ± 0.0226	0.1840 ± 0.0090	0.0757 ± 0.0038	0.4458 ± 0.0102
BSVM-V	0.0122 ± 0.0001	0.5063 ± 0.0161	0.2149 ± 0.0092	0.0885 ± 0.0033	0.4038 ± 0.0088

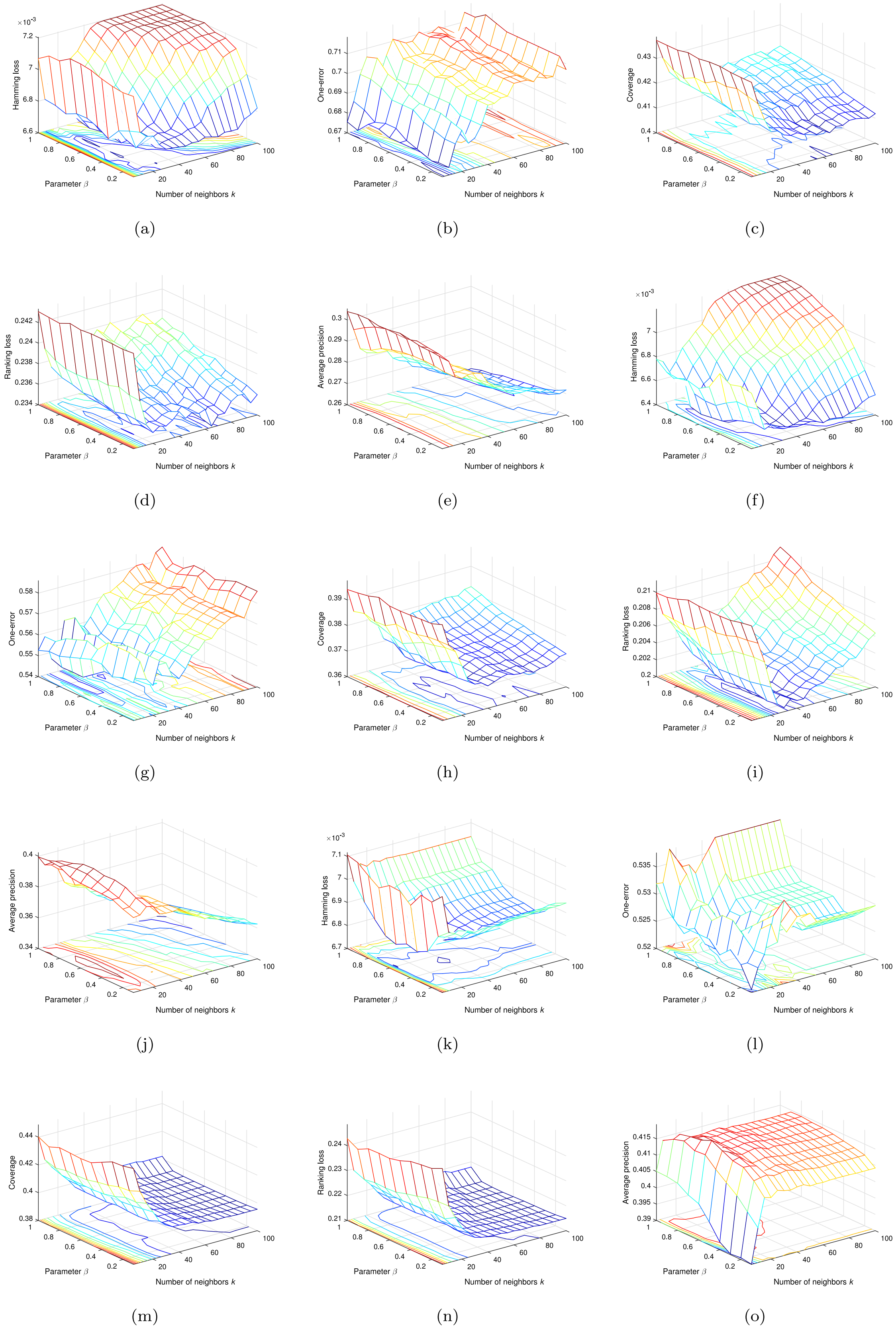
From Table 2, we can observe that MLLC is superior to BSVM. To be specific, an interesting observation can be made that the performance of MLLC has minimal improvement compared with BSVM in terms of *hamming loss*, but MLLC significantly outperforms BSVM in terms of *coverage*, *ranking loss*, and *average precision*. Considering the metric of *one-error*, we can see that MLLC achieves better performance on 6 of 8 subtasks. In light of that BSVM is a degenerated version of MLLC, we can conclude that MLLC is effective in capturing the label correlations. For further verifying the effective of MLLC, LIFT is used to replace BSVM for exploring an improved LIFT, and the experimental result is shown in Table 3. From Table 3, we can see that MLLC achieves better performance than LIFT in most cases, which is proved to be a feasible scheme to exploit the label correlations based on a base multi-label classifier.

In short, we can draw a conclusion that the proposed method can improve the performance of multi-label methods. Therefore, we set parameter α as a value in less than 0.5 to exploit the label correlations for performance improvement.

4.4. Multi-source fusion

We further validate the proposed multi-source fusion method for multi-label learning, namely MLSO, and use the above three data sets to launch the test.

As a preliminary endeavor, we estimate the average number of positive predictions, which are generated by MLLC, based on each data source from DR, TCM, and Corel5k data sets, as shown in Figs. 3(a)–(c) respectively. From Fig. 3(a), we can observe that



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

(m) (n) (o)

Fig. 4. Sensitivity analysis of β and k. CHST: (a) *hamming loss*, (b) *one-error*, (c) *coverage*, (d) *ranking loss*, and (e) *average precision*. SIEF: (f) *hamming loss*, (g) *one-error*, (h) *coverage*, (i) *ranking loss*, and (j) *average precision*. PRTA: (k) *hamming loss*, (l) *one-error*, (m) *coverage*, (n) *ranking loss*, and (o) *average precision*.

all data sources from DR data set result in a balanced result considering the average number of positive predictions. However, the quality of data sources from TCM data set is different. As shown in Fig. 3(b), for multi-label classification, the generated result based on SYMP data source amounts to more than 200 positive predictions, whereas the other data sources, i.e. THSV, TLAB, and TRGB, are *small* sources which possess a noticeably weaker class-discriminative ability. As for Corel5k data set, we can infer from Fig. 3(c) that DESI and RGB data sources have stronger class-discriminative ability than GIST and HAHU.

Based on the three multi-source multi-label data sets, we first verify the feasibility and effectiveness of MLSO. Tables 4–6 summarize the fusion results on DR, TCM, and Corel5k data sets respectively. From Tables 4 and 6, we can observe that on DR and Corel5k data sets, MLSO always obtains the best performance by combining all data sources, and the predicted result based on single data source compares unfavorably with the fusion result. On TCM data set, we can see from Table 5 that MLSO has the best result with respect to *hamming loss*, *one-error*, and *average precision* by fusing all the sources, hence we can conclude that the *small* sources do contribute to the performance improvement, whereas the fusion result is a little inferior to the predicted result based on SYMP in terms of *coverage* and *ranking loss*.

Furthermore, we compare MLSO with some common multi-source fusion strategies. Tables 7–9 show the comparison results on DR, TCM, and Corel5k data sets respectively, in which MLLC-A and MLLC-V stand for the average result and voting result generated by MLLC from multiple sources respectively; BSVM-A and BSVM-V stand for the average result and voting result generated by BSVM from multiple sources respectively. From Tables 7–9, we have a couple of observations. Compared with MLLC-A and BSVM-A, MLSO can obtain better fusion results on all the data sets, whereas on Corel5k data set, MLSO is inferior to the Averaging method with respect to *one-error*, such as MLLC-A and BSVM-A. Similarly, MLSO is superior to the Major Voting method on the three data sets. In addition, we can see from Tables 7 and 9 that the fusion method of Averaging has good performance on DR and Corel5k data sets, such as MLLC-A. Nevertheless, the Averaging method cannot deal with long-tail data properly, and we can see from Table 8 that the performance of MLLC-A and BSVM-A is poor on TCM data set.

According to the experimental results of data source comparison and method comparison, we can come to the conclusion that the proposed method is feasible and effective in multi-source fusion.

4.5. Parameter sensitivity analysis

α, β, and k are three parameters involved in the algorithm optimization. Among these parameters, α is a parameter to reflect the influence of label correlations. In view of exploiting the label correlations which helps to the performance improvement (as shown in Section 4.3), α tends to assign a value in less than 0.5. Definitely, in this paper, α is set to be 0.3. In addition, for analyzing the sensitivity of the proposed method with respect to β and k, we conduct the experiment on DR data set, as shown in Fig. 4. We can observe from Fig. 4 that the performance of the proposed method is going to change dramatically regarding each evaluation metric while k is varied from 10 to 100, and the performance is also sensitive to β. For example, we can see from Fig. 4(j) that the highest *average precision* based on SIEF is achieved at some intermediate values of β, in the similar way, the best result on *hamming loss* and *average precision* can be found out based on PRTA, as shown in Figs. 4(k) and 4(o).

4.6. Convergence analysis

As stated in Section 3.3, the convergence of the proposed algorithm is guaranteed, and the estimation of the two unknown variable sets, namely the distribution of source weights and the final prediction, is the crucial link to study the convergence rate. Once the estimated distribution of source weights converges, the final prediction becomes stable. We display the change of the objective value with respect to each iteration on DR data set, as shown in Fig. 5. From Fig. 5, we can observe that the objective decreases fast in a few iterations and then reaches a stable stage, which is effective in obtaining the optimal solution.

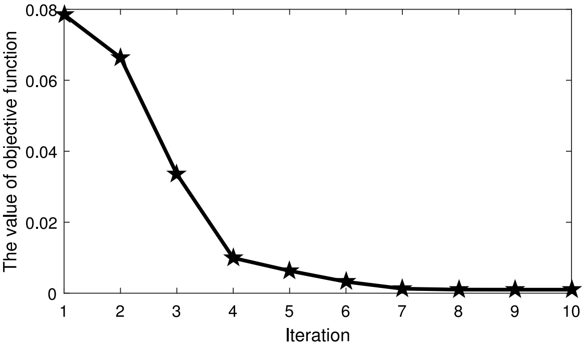


Fig. 5. Convergence analysis. Loss of MLSO w.r.t. the number of iterations on DR data set.

5. Conclusion

In this paper, we proposed a unified multi-source-based optimization framework for multi-label learning. Concretely, we designed an optimization objective function to achieve the weighted combination prediction by incorporating the label correlations estimation. The experiments on three real-world data sets showed that the effectiveness in exploiting the label correlations and learning a consensus-based classification result from multi-source data, especially long-tail data. A comparative study with some other methods manifested a competitive performance of the proposed method in multi-label classification and multi-source fusion.

In future work, it is interesting to propose other multi-source multi-label learning approaches, such as using graphs as a medium for the algorithm design. In addition, we will also pay attention to design interactive machine learning method [58] for the joint learning of multiple labels and data sources.

Acknowledgments

This work is supported by grants from the National Natural Science Foundation of China (No. 61572409, No. U1705286 & No. 61571188), Fujian Province 2011 Collaborative Innovation Center of TCM Health Management, China and Collaborative Innovation Center of Chinese Oolong Tea Industry-Collaborative Innovation Center (2011) of Fujian Province, China, Fund for Integration of Cloud Computing and Big Data, Innovation of Science and Education, China.

References

[1] A. Labrinidis, H.V. Jagadish, Challenges and opportunities with big data, PVLDB 5 (12) (2012) 2032–2033.

[2] X. Wu, X. Zhu, G. Wu, W. Ding, Data mining with big data, IEEE Trans. Knowl. Data Eng. 26 (1) (2014) 97–107.

[3] O. Chapelle, P. Haffner, V. Vapnik, Support vector machines for histogram-based image classification, IEEE Trans. Neural Netw. 10 (5) (1999) 1055–1064.

[4] J. Peng, C. Tang, D. Yang, J. Zhang, J. Hu, Similarity computing model of high dimension data for symptom classification of Chinese traditional medicine, Appl. Soft Comput. 9 (1) (2009) 209–218.

[5] E. Gibaja, S. Ventura, A tutorial on multilabel learning, ACM Comput. Surv. 47 (3) (2015) 52:1–52:38.

[6] M. Zhang, Z. Zhou, A review on multi-label learning algorithms, IEEE Trans. Knowl. Data Eng. 26 (8) (2014) 1819–1837.

[7] Y. Lin, Q. Hu, J. Zhang, X. Wu, Multi-label feature selection with streaming labels, Inform. Sci. 372 (2016) 256–275.

[8] Y. Lin, Q. Hu, J. Liu, J. Chen, J. Duan, Multi-label feature selection based on neighborhood mutual information, Appl. Soft Comput. 38 (2016) 244–256.

[9] J. Zhang, C. Li, D. Cao, Y. Lin, S. Su, L. Dai, S. Li, Multi-label learning with label-specific features by resolving label correlations, Knowl.-Based Syst. 159 (2018) 148–157.

[10] P. Hou, X. Geng, M. Zhang, Multi-label manifold learning, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence, 2016, pp. 1680–1686.

[11] T. Li, F. de la Prieta Pintado, J.M. Corchado, J. Bajo, Multi-source homogeneous data clustering for multi-target detection from cluttered background with misdetection, Appl. Soft Comput. 60 (2017) 436–446.

[12] J. Li, D. Zhang, Y. Li, J. Wu, B. Zhang, Joint similar and specific learning for diabetes mellitus and impaired glucose regulation detection, Inform. Sci. 384 (2017) 191–204.

[13] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, J. Han, Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation, in: Proceedings of the International Conference on Management of Data, 2014, pp. 1187–1198.

[14] J. Zhang, C. Li, Y. Lin, Y. Shao, S. Li, Computational drug repositioning using collaborative filtering via multi-source fusion, Expert Syst. Appl. 84 (2017) 281–289.

[15] S. Xie, X. Kong, J. Gao, W. Fan, P.S. Yu, Multilabel consensus classification, in: Proceedings of the 13th International Conference on Data Mining, 2013, pp. 1241–1246.

[16] J. Xu, V. Jagadeesh, B.S. Manjunath, Multi-label learning with fused multimodal bi-relational graph, IEEE Trans. Multimedia 16 (2) (2014) 403–412.

[17] C. Shi, X. Kong, P.S. Yu, B. Wang, Multi-label ensemble learning, in: Lecture Notes in Artificial Intelligence, vol. 6913, 2011, pp. 223–239.

[18] X. Zhang, Q. Yuan, S. Zhao, W. Fan, W. Zheng, Z. Wang, Multi-label classification without the multi-label cost, in: Proceedings of the SIAM International Conference on Data Mining, 2010, pp. 778–789.

[19] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, A.Y. Ng, Multimodal deep learning, in: Proceedings of the 28th International Conference on Machine Learning, 2011, pp. 689–696.

[20] N. Srivastava, R. Salakhutdinov, Multimodal learning with deep Boltzmann machines, in: Advances in Neural Information Processing Systems, vol. 25, 2012, pp. 2231–2239.

[21] R.V. Devi, S.S. Sathya, M.S. Coumar, Evolutionary algorithms for de novo drug design — A survey, Appl. Soft Comput. 27 (2015) 543–552.

[22] P. Zhang, P. Agarwal, Z. Obradovic, Computational drug repositioning by ranking and integrating multiple data sources, in: Lecture Notes in Computer Science, vol. 8190, 2013, pp. 579–594.

[23] P. Gu, H. Chen, Modern bioinformatics meets traditional Chinese medicine, Brief. Bioinform. 15 (6) (2014) 984–1003.

[24] L. Dai, J. Zhang, C. Li, C. Zhou, S. Li, Multi-label feature selection with application to TCM state identification, Concurrency Computat. Pract. Exper. https://doi.org/10.1002/cpe.4634.

[25] G. Liu, G. Li, Y. Wang, Y. Wang, Modelling of inquiry diagnosis for coronary heart disease in traditional Chinese medicine by using multi-label learning, BMC Complement. Altern. Med. 10 (2010) 37.

[26] W. Wu, J. Liu, H. Chang, Latent class model based diagnostic system utilizing traditional Chinese medicine for patients with systemic lupus erythematosus, Expert Syst. Appl. 38 (1) (2011) 281–287.

[27] M.R. Boutell, J. Luo, X. Shen, C.M. Brown, Learning multi-label scene classification, Pattern Recognit. 37 (9) (2004) 1757–1771.

[28] M. Zhang, L. Wu, LIFT: Multi-label learning with label-specific features, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 107–120.

[29] J. Fürnkranz, E. Hüllermeier, E.L. Mencía, K. Brinker, Multilabel classification via calibrated label ranking, Mach. Learn. 73 (2) (2008) 133–153.

[30] J. Huang, G. Li, Q. Huang, X. Wu, Learning label-specific features and class-dependent labels for multi-label classification, IEEE Trans. Knowl. Data Eng. 28 (12) (2016) 3309–3323.

[31] Z. Sun, J. Zhang, L. Dai, C. Li, C. Zhou, J. Xin, S. Li, Mutual information based multi-label feature selection via constrained convex optimization, Neurocomputing, https://doi.org/10.1016/j.neucom.2018.10.047.

[32] J. Read, B. Pfahringer, G. Holmes, E. Frank, Classifier chains for multi-label classification, Mach. Learn. 85 (3) (2011) 333–359.

[33] G. Tsoumakas, I.P. Vlahavas, Random k -labelsets: An ensemble method for multilabel classification, in: Proceedings of the 18th European Conference on Machine Learning, 2007, pp. 406–417.

[34] S. Huang, Z. Zhou, Multi-label learning by exploiting label correlations locally, in: Proceedings of the 26th AAAI Conference on Artificial Intelligence, 2012, pp. 945–955.

[35] S. Clinchant, J. Ah-Pine, G. Csurka, Semantic combination of textual and visual information in multimedia retrieval, in: Proceedings of the 1st International Conference on Multimedia Retrieval, 2011, p. 44.

[36] C. Snoek, M. Worring, A.W.M. Smeulders, Early versus late fusion in semantic video analysis, in: Proceedings of the 13th ACM International Conference on Multimedia, 2005, pp. 399–402.

[37] M.J. Huiskes, B. Thomee, M.S. Lew, New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative, in: Proceedings of the 11th International Conference on Multimedia Information Retrieval, 2010, pp. 527–536.

[38] J. Bleiholder, F. Naumann, Data fusion, ACM Comput. Surv. 41 (1) (2008) 1:1–1:41.

[39] Z. Jiang, A decision-theoretic framework for numerical attribute value reconciliation, IEEE Trans. Knowl. Data Eng. 24 (7) (2012) 1153–1169.

[40] C. Xu, D. Tao, C. Xu, A survey on multi-view learning, CoRR abs/1304.5634, http://arxiv.org/abs/1304.5634.

[41] M. Zitnik, B. Zupan, Data fusion by matrix factorization, IEEE Trans. Pattern Anal. Mach. Intell. 37 (1) (2015) 41–53.

[42] A. Clauset, C.R. Shalizi, M.E.J. Newman, Power-law distributions in empirical data, SIAM Rev. 51 (4) (2009) 661–703.

[43] E. Mustafaraj, S. Finn, C. Whitlock, P.T. Metaxas, Vocal minority versus silent majority: Discovering the opionions of the long tail, in: Proceedings of the 3rd International Conference on Social Computing, 2011, pp. 103–110.

[44] M. Wang, X. Hua, X. Yuan, Y. Song, L. Dai, Optimizing multi-graph learning: Towards a unified video annotation scheme, in: Proceedings of the 15th International Conference on Multimedia, 2007, pp. 862–871.

[45] Z. Fu, H.H. Ip, H. Lu, Z. Lu, Multi-modal constraint propagation for heterogeneous image clustering, in: Proceedings of the 19th International Conference on Multimedia, 2011, pp. 143–152.

[46] J. Gao, F. Liang, W. Fan, Y. Sun, J. Han, A graph-based consensus maximization approach for combining multiple supervised and unsupervised models, IEEE Trans. Knowl. Data Eng. 25 (1) (2013) 15–28.

[47] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, J. Han, A confidence-aware approach for truth discovery on long-tail data, PVLDB 8 (4) (2014) 425–436.

[48] M.A. Domingues, F.G.A.M. Jorge, J.P. Leal, J. Vinagre, L. Lemos, M. Sordo, Combining usage and content in an online music recommendation system for music in the long-tail, in: Proceedings of the 21st World Wide Web Conference, 2012, pp. 925–930.

[49] C. Chang, C. Lin, LIBSVM: A library for support vector machines, ACM Trans. Inf. Syst. 2 (3) (2011) 27:1–27:27.

[50] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: Proceedings of the 18th International Conference Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241.

[51] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.

[52] D.S. Wishart, C. Knox, A. Guo, D. Cheng, S. Shrivastava, D. Tzur, B. Gautam, M. Hassanali, DrugBank: A knowledgebase for drugs, drug actions and drug targets, Nucleic Acids Res. 36 (2008) 901–906.

[53] O. Bodenreider, The unified medical language system (UMLS): Integrating biomedical terminology, Nucleic Acids Res. 32 (2004) 267–270.

[54] Y. Wang, J. Xiao, T.O. Suzek, J. Zhang, J. Wang, S.H. Bryant, PubChem: A public information system for analyzing bioactivities of small molecules, Nucleic Acids Res. 37 (2009) 623–633.

[55] M. Kuhn, M. Campillos, I. Letunic, L.J. Jensen, P. Bork, A side effect resource to capture phenotypic effects of drugs, Mol. Syst. Biol. 6 (2010) 343.

[56] M. Guillaumin, T. Mensink, J. Verbeek, C. Schmid, TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation, in: Proceedings of the 12th International Conference on Computer Vision, 2009, pp. 309-316.

[57] X. Wu, Z. Zhou, A unified view of multi-label performance measures, in: Proceedings of the 34th International Conference on Machine Learning, 2017, pp. 3780–3788.

[58] A. Holzinger, Interactive machine learning for health informatics: when do we need the human-in-the-loop? Brain Inform. 3 (2) (2016) 119–131.