



Multi-label learning with missing and completely unobserved labels

Jun Huang¹ · Linchuan Xu² · Kun Qian³ · Jing Wang⁴ · Kenji Yamanishi⁵

Received: 30 April 2020 / Accepted: 17 February 2021 / Published online: 12 March 2021
© The Author(s) 2021

Abstract

Multi-label learning deals with data examples which are associated with multiple class labels simultaneously. Despite the success of existing approaches to multi-label learning, there is still a problem neglected by researchers, i.e., not only are some of the values of observed labels missing, but also some of the labels are completely unobserved for the training data. We refer to the problem as *multi-label learning with missing and completely unobserved labels*, and argue that it is necessary to discover these completely unobserved labels in order to mine useful knowledge and make a deeper understanding of what is behind the data. In this paper, we propose a new approach named MCUL to solve multi-label learning with Missing and Completely Unobserved Labels. We try to discover the unobserved labels of a multi-label data set with a clustering based regularization term and describe the semantic meanings of them based on the label-specific features learned by MCUL, and overcome the problem of missing labels by exploiting label correlations. The proposed method MCUL can predict both the observed and newly discovered labels simultaneously for unseen data examples. Experimental results validated over ten benchmark datasets demonstrate that the proposed method can outperform other state-of-the-art approaches on observed labels and obtain an acceptable performance on the new discovered labels as well.

Keywords Multi-label learning · Missing labels · Completely unobserved labels · Unseen labels · Discovering new labels

1 Introduction

1.1 Background and motivation

Multi-label learning (Gibaja and Ventura 2015; Herrera et al. 2016; Tsoumakas et al. 2010; Zhang and Zhou 2014) is a learning framework for learning in the presence of

Responsible editor: Grigorios Tsoumakas.

Extended author information available on the last page of the article

label ambiguity, where each instance can be associated with multiple possible class labels simultaneously. Many well-established approaches have been proposed, such as (Chu et al. 2019; Decubber et al. 2019; Liu 2019; Liu and Shen 2019; Masera and Blanzieri 2019; Nguyen and Hüllermeier 2019; Park and Read 2019; Huang et al. 2018; Wydmuch et al. 2018; Zhang and Wu 2019). In multi-label learning, a common assumption is that all the class labels and their values are observed before the training process. However, in some real applications, not only *are some of the values of the observed labels missing*, but also *some of the labels are completely unobserved for the training data*. We summarized three possible reasons as follows.

1. *The labeling process is complex and costly* In the labeling process of multi-label learning, a set of possible labels from a target set will be annotated for each data example. This stage is very complex and time-consuming, especially for a large-scale data set with millions of labels (Bhatia et al. 2016). It is inevitable to induce errors and missing values, and even result in some labels totally unlabeled for all the related data examples.
2. *Some labels are intentionally omitted* For example, in image annotation, people may be only interested in the main objects of an image, and the background of image may not be annotated, such as *grass* and *land*. However, in (Pham et al. 2015), it has been proved that the performance on observed labels can be improved by discovering these labels.
3. *Some labels are unknown* For example, in disease diagnosis, complicated diseases may exist but are unknown due to the limitation of human's knowledge or the shortage of examination (Zhang et al. 2018, 2020).

There are several lines of study that are related to the problem proposed in this paper. In Fig. 1, we illustrate the differences between the learning scenario proposed in the paper and other previous related learning problems, i.e., multi-label learning with missing labels, and online or class-incremental learning. The detailed discussions and analyses now follow.

First, in multi-label learning with missing labels, all the class labels are known in advance, whereas some of labelling results are missing or unobserved. A lot of approaches have been proposed for multi-label learning with missing labels, such as (Huang et al. 2019; Sun et al. 2010; Xu et al. 2013; Yu et al. 2014; Zhu et al. 2018). However, to successfully apply these approaches, *one essential precondition is that each label has at least one positive data example*. The problem setting on this precondition is different from that of multi-label learning with completely unobserved labels.

Second, class incremental or online learning approaches (Da et al. 2014; Mu et al. 2017; Qu et al. 2009; Zhu et al. 2018) can handle classification with novel labels which are unseen in the training stage but appear in the test stage. The novel labels are unobserved because of the corresponding data examples are unobserved. By contrast, in our problem setting, *novel labels are unobserved but the data examples are observed*. Moreover, in multi-label learning, *novel labels may not be mutually exclusive with existing observed labels, but have correlations with each other*. Therefore, these approaches can not be applied to multi-label learning with completely unobserved labels.

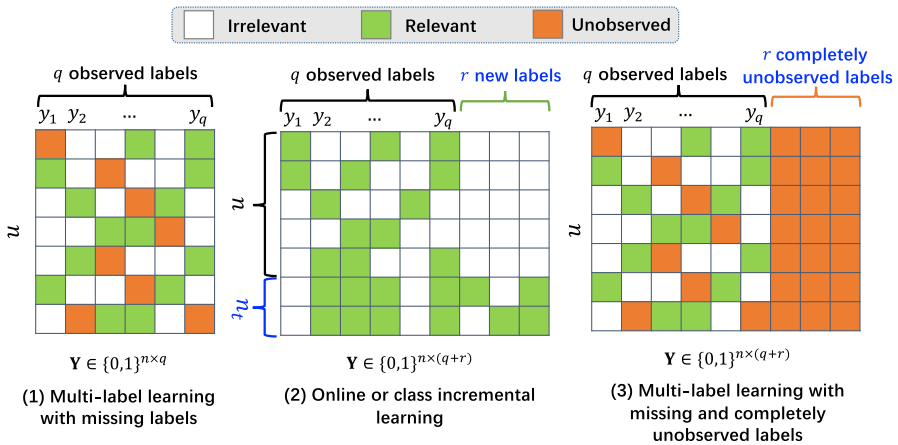


Fig. 1 Differences between previous related learning problems

The problem of detecting unobserved labels has been studied under single-instance single-label learning (Zhang et al. 2020) and multi-instance multi-label learning (Pham et al. 2015; Zhu et al. 2017). In single-instance single-label learning, unobserved labels are mutually exclusive with each other including the observed ones. In multi-instance multi-label learning, each data example is represented by multiple instances. Different from these two problems, for the proposed problem, each data example is represented by a single instance and associated with multiple class labels (including the unobserved labels) simultaneously which may have correlations with each other. In addition, these approaches can not handle missing values of the observed labels.

We refer to the proposed problem as multi-label learning with missing and completely unobserved labels, and introduce a formal definition of it as follows.

Definition 1 (Multi-label learning with missing and completely unobserved labels.)

For a given multi-label learning dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, $\mathcal{X} \in \mathbb{R}^d$ indicates the feature space, and $\mathcal{Y} = \{y_1, \dots, y_q, y_{q+1}, \dots, y_{q+r}\}$ represents the full label space. In the training stage, the first q labels are observed and the rest r labels are completely unobserved. For the q observed labels, some of the annotation results are missing, but each label has at least one positive data example. While for the r completely unobserved labels, the semantic meanings of them and their labelling results for the n data examples are totally unknown.

The task of multi-label learning with missing and completely unobserved labels is to build a robust multi-label classification model which can discover previously unobserved labels and overcome the problem of missing values of the observed labels in the dataset. Meanwhile, the model can predict both the observed and unobserved labels simultaneously for unseen data examples. Besides, it would be better, if the meanings of unobserved labels can be interpreted.

1.2 Significance and contribution

Properly modeling the unobserved labels in multi-label learning can have positive impacts from two aspects. First, it enables effective discovering of unobserved labels and makes a deeper understanding of what is behind the multi-label data. Second, by discovering and making good use of the information of the unobserved labels in the multi-label data, we can better build a robust classification model for the observed labels and improve the accuracy of prediction.

In this paper, we propose a novel approach named MCUL to solve multi-label learning with Missing and Completely Unobserved Labels. MCUL is a robust multi-label classification model which can discover the completely unobserved labels and overcome the problem of partially missing values of the observed labels. In the test stage, it can predict unseen data examples with both the observed and unobserved labels simultaneously. The contributions of this paper are summarized as follows.

- We introduce the problem of multi-label learning with missing and completely unobserved labels. To the best of our knowledge, this topic is firstly addressed in multi-label learning.
- We propose a new approach named MCUL for the proposed problem, where a clustering-based regularization term is utilized to discover the unobserved labels, and label correlations are exploited to overcome the problem of missing values for both the observed and new discovered unobserved labels. We try to describe the semantic meaning of the new discovered labels based on the label-specific features which are learned by MCUL.
- We present three new evaluation metrics for the evaluation on the completely unobserved labels. Since the one-to-one correspondences between the ground-truth and new discovered labels are unknown, the existing evaluation metrics can not be applied directly. We propose to evaluate the results for the labels which are best matched based on some existing evaluation metrics for multi-label learning, such as *ranking loss* and *coverage*.

The advantages of the proposed framework are demonstrated by experiments on observed label prediction and novel label discovering over ten real multi-label datasets. The performance on observed labels can be improved by discovering and modeling the completely unobserved labels. The label-specific features with high weights have a strong semantic correlation with the name of the best-matched labels, and can be used to describe the semantic meaning for the new discovered labels.

2 Related work

Multi-label learning (Gibaja and Ventura 2015; Herrera et al. 2016; Tsoumakas et al. 2010; Zhang and Zhou 2014) deals with data examples which are associated with multiple class labels simultaneously. In the past decades, many advanced approaches have been proposed to solve interesting problems in multi-label learning.

According to the popular taxonomy firstly proposed in (Tsoumakas et al. 2010), existing multi-label learning approaches can mainly be divided into two categories:

problem transformation (PT) strategy and algorithm adaption (AA) strategy. For the problem transformation strategy, a multi-label classification problem is transformed into one or more single-label classification problems that can be solved with a single-label classification algorithm, such as (Boutell et al. 2004; Dembczyński et al. 2010; Read et al. 2008, 2009; Tsoumakas et al. 2011). For the algorithm adaption strategy, traditional single-label classification algorithms are extended to solve multi-label classification problems directly, such as (Elisseff and Jason 2001; Fürnkranz et al. 2008; Zhang and Zhou 2006, 2007). Nevertheless, existing approaches mainly assume that all the class labels are observed before the training process and the set of target labels is a closed set. Although the success has been made by existing studies on multi-label learning, there is still a challenging problem that some of the class labels are completely unobserved during the training stage. There are several lines of study that are related to the problem we proposed in this paper, such as multi-label learning with missing labels, class-incremental learning and stream multi-label learning.

Many approaches have been proposed for multi-label learning with missing labels, and can be mainly grouped into two categories. One strategy is to recover a full label matrix based on the matrix completion or factorization techniques by exploiting label or instance correlations, such as (Huang et al. 2019; Xu et al. 2013; Zhu et al. 2018). Another strategy is assuming that we have known which entries are missing, and then to calculate the classification loss without considering them, such as (Sun et al. 2010; Tan et al. 2018; Yu et al. 2014). The essential precondition for these two strategies is that each label has at least one positive data example. Nevertheless, these two strategies both will not work if one label is completely unobserved.

Some approaches have been proposed for class-incremental learning (Da et al. 2014; Shi et al. 2014) and stream multi-label learning (Mu et al. 2017; Qu et al. 2009; Read et al. 2011; Zhu et al. 2018). In these two problems, new labels are unobserved during the training stage, but appear in the test stage. The labels are unobserved because the corresponding data examples are also unobserved during the training stage. While in our problem, the data examples are observed but some labels are completely unobserved during the training stage. In addition, for class-incremental learning and stream classification problems, if one label is unobserved in the training stage and does not appear in the test stage, it will never be discovered.

There are several highly related studies with the purpose of discovering unobserved labels for the training data. ExML (Zhang et al. 2020) assumes that the unobserved labels are wrongly annotated as observed labels, and examines and investigates the training data set by actively augmenting the feature space to discover potentially unobserved labels. However, it can not be applied to multi-label learning, and the problem setting is also different from us. MIMLNC (Pham et al. 2015) is a probabilistic model to identify novel instances for multi-instance multi-label learning, and it assumes that all novel instances belong to a single new label. DMNL (Zhu et al. 2017) assumes that there are k unobserved labels, and tries to discover multiple novel labels for multi-instance multi-label learning with a clustering based regularization term. These two approaches are hardly applied to general single-instance multi-label learning.

By surveying previous studies on multi-label learning, it is found that none of existing approaches can directly address the potential problem of multi-label learning with missing and completely unobserved labels. In this paper, we propose a novel

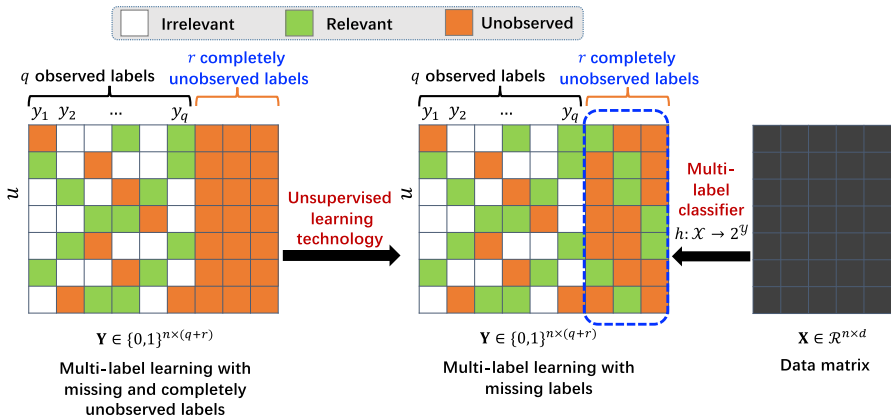


Fig. 2 The framework of the proposed method MCUL

approach named MCUL which can discover the completely unobserved labels and overcome the problem of partially missing values of the observed labels, and predict both the observed and unobserved labels simultaneously to the unseen data examples in the test stage.

3 The proposed approach

3.1 Preliminary

To describe the new problem settings given in definition 1, we provide the following formal notations.

$\mathcal{X} \in \mathbb{R}^d$ indicates the d -dimensional feature space, and $\mathcal{Y} = \{y_1, \dots, y_q\}$ represents the label set of q observed labels. Assuming there are r different unobserved labels which are indicated by $\tilde{\mathcal{Y}} = \{y_{q+1}, \dots, y_{q+r}\}$. As a result, there are l labels totally, and the complete label set will be $\hat{\mathcal{Y}} = \mathcal{Y} \cup \tilde{\mathcal{Y}} = \{y_1, \dots, y_q, y_{q+1}, \dots, y_l\}$, where $l = q + r$. $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n]^T \in \mathbb{R}^{n \times d}$ is used to indicate the data matrix of a multi-label learning training set, and $\hat{\mathbf{Y}} = [\mathbf{Y}, \tilde{\mathbf{Y}}] \in \{0, 1\}^{n \times l}$ is used to indicate the full label matrix. Here, $\mathbf{Y} \in \{0, 1\}^{n \times q}$ indicates the label matrix for the q observed labels and some entries of it are missing. If $y_{ij} = 1$, it indicates that the \mathbf{x}_i belongs to y_j . If $y_{ij} = 0$, it indicates that the \mathbf{x}_i does not belong to y_j or the value is missing. $\tilde{\mathbf{Y}} \in \{0, 1\}^{n \times r}$ indicates the label matrix for the unobserved labels, and all the entries of it are missing during the training stage, i.e., $\tilde{y}_{ij} = 0, \forall 1 \leq i \leq n, 1 \leq j \leq r$.

For the proposed problem, we aim to construct a robust multi-label learning model $h : \mathcal{X} \rightarrow 2^{\hat{\mathcal{Y}}}$ which can predict unseen data examples with both the observed and unobserved labels simultaneously. In this paper, we propose a new method MCUL to solve multi-label learning with Missing and Completely Unobserved Labels. The framework is shown in Fig. 2. The main idea is that we first transform the label matrix from completed missing to partially missing with the help of unsupervised learning

techniques, and then we learn a model from the feature space to the augmented label space and try to recover the missing entries by exploiting label correlations. Specifically, MCUL is composed of two parts, i.e., *discovering the completely unobserved labels* and *building a robust multi-label learning classifier for observed and unobserved labels*.

3.2 Discovering the completely unobserved labels

To construct a multi-label classification model $h : \mathcal{X} \rightarrow 2^{\hat{\mathcal{Y}}}$, we need the full label matrix $\hat{\mathbf{Y}} = [\mathbf{Y}, \bar{\mathbf{Y}}]$ for the training data. However, $\bar{\mathbf{Y}}$ is completely unobserved and unknown during the training stage. Therefore, we need resort to some unsupervised learning techniques, such as *clustering*. In Ding et al. (2005), it is indicated that the nonnegative matrix factorization (NMF) factorizing a symmetric similarity matrix \mathbf{S} into $\mathbf{H}\mathbf{H}^\top$ is equivalent to the soft k -means clustering. The optimization objective function of it is formulated as

$$\min_{\mathbf{H}} \|\mathbf{S} - \mathbf{H}\mathbf{H}^\top\|_F^2, \quad s.t. \mathbf{H} \geq 0, \quad (1)$$

where $\mathbf{S} \in \mathbb{R}^{n \times n}$ is the similarity matrix containing pairwise similarities or the kernels, and $\mathbf{H} \in \mathbb{R}^{n \times l}$ is the clustering indicator matrix. For a matrix \mathbf{A} , $\|\mathbf{A}\|_F$ indicates the Frobenius norm of it, and $\|\mathbf{A}\|_F^2 = \text{tr}(\mathbf{A}^\top \mathbf{A})$.

For the proposed problem, we have already obtained the labeling result for part of the labels, i.e., $\mathbf{Y} \in \mathbb{R}^{n \times q}$ is known in advance. Therefore, part of the results of \mathbf{H} should be consistent with \mathbf{Y} . It is noted that $\mathbf{h}_i \mathbf{h}_j^\top = \sum_{m=1}^l h_{im} h_{jm}$, where \mathbf{h}_i indicates the i -th row of \mathbf{H} . Changing of the order of the columns of \mathbf{H} will not change the value of $\mathbf{H}\mathbf{H}^\top$. Without loss of generality, we assume that the results of the first q columns of \mathbf{H} should be consistent with that of the q observed labels. Consequently, we extend the problem (1) to the following one

$$\min_{\mathbf{H}} \|\mathbf{S} - \mathbf{H}\mathbf{H}^\top\|_F^2, \quad s.t. \mathbf{H}\mathbf{P} = \mathbf{Y}, \mathbf{H} \in [0, 1]^{n \times l}, \quad (2)$$

where $\mathbf{P} \in \{0, 1\}^{l \times q}$ is a projection matrix with ones on the main diagonal and zeros elsewhere.

In this paper, the similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$ is calculated by the Gaussian kernel based on the feature and label spaces simultaneously. Each element s_{ij} is defined as

$$s_{ij} = \exp\left(\frac{-\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2^2}{2\sigma^2}\right), \quad (3)$$

where $\hat{\mathbf{x}}_i = [\mathbf{x}_i, \mathbf{y}_i]$, and σ is set to be 1 in the experiment.

3.3 Building a robust multi-label learning classifier

After obtaining the preliminary labeling results of the r unobserved labels, we can construct a multi-label classifier for both of the q observed and r completely unobserved labels simultaneously. Here, we learn a linear model for $h : \mathcal{X} \rightarrow 2^{\hat{\mathcal{Y}}}$, then the optimization problem becomes

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{H}} \quad & \frac{1}{2} \|\mathbf{X}\mathbf{W} + \mathbf{1}_n \mathbf{b}^\top - \mathbf{H}\|_F^2 + \frac{\lambda_0}{4} \|\mathbf{S} - \mathbf{H}\mathbf{H}^\top\|_F^2 \\ \text{s.t.} \quad & \mathbf{H}\mathbf{P} = \mathbf{Y}, \mathbf{H} \in [0, 1]^{n \times l}, \end{aligned} \tag{4}$$

where $\mathbf{W} \in \mathbb{R}^{d \times l}$ is the model coefficient matrix, $\mathbf{b} \in \mathbb{R}^l$ is the bias, and $\mathbf{1}_n$ denotes the vector of size n . For simplicity, the bias \mathbf{b} can be absorbed into \mathbf{W} by adding an additional feature with all the values equal to 1 for the data matrix, i.e., $\mathbf{X} = [\mathbf{X}, \mathbf{1}_n]$.

As mentioned in the previous section, \mathbf{Y} is observed but with some missing entries. While the problem (1) is not designed for multi-label learning, and thus there will be missing entries in \mathbf{H} as well. In the problem (4), we have tried to recover the full label matrix by exploiting the instance similarity, i.e., if two data instances \mathbf{x}_i and \mathbf{x}_j are similar in the feature space \mathcal{X} , their label vectors \mathbf{h}_i and \mathbf{h}_j will similar in the label space $\hat{\mathcal{Y}}$. On the other hand, we can resort to reconstruct the missing entries from the results of other labels by exploiting the label similarity. From the perspective of the similarity of label, the assignment of one certain label to training instances can be reconstructed from other labels, especially from its highly similar labels. The fourth term of (5) is adopted to model label reconstruction, i.e., $h_{ij} \approx \sum_{m=1}^l h_{im} c_{mj}$, where $\mathbf{C} \in \mathbb{R}^{l \times l}$ represents the reconstruction coefficient matrix, and each element c_{ij} indicates the reconstruction coefficient that label y_j is derived from y_i .

In addition, we can reconstruct the missing entries in \mathbf{H} by modeling label correlations. In particular, we hope that highly correlated labels have similar outputs. Specifically, if two labels y_i and y_j have a strong correlation, then their model parameters \mathbf{w}_i and \mathbf{w}_j will be similar, and thus the distance (i.e., $\|\mathbf{w}_i - \mathbf{w}_j\|_2^2$) should be small. Otherwise, the distance should be large. Since all the binary classifiers for each label have the same input data \mathbf{X} , if labels y_i and y_j are highly correlated, their corresponding classifiers will have similar outputs by adding the constraint. The fifth term of (5) is adopted to model pairwise label correlation including both observed and unobserved labels, where \mathbf{L} represents the graph Laplacian matrix of the label correlation matrix which is calculated by cosine similarity between label pairs of \mathbf{H} . Consequently, the objective function can be rewritten as

$$\begin{aligned} \min_{\mathbf{W}, \mathbf{C}, \mathbf{H}} \quad & \frac{1}{2} \|\mathbf{X}\mathbf{W} - \mathbf{H}\mathbf{C}\|_F^2 + \frac{\lambda_0}{4} \|\mathbf{S} - \mathbf{H}\mathbf{H}^\top\|_F^2 + \frac{\lambda_1}{2} \|\mathbf{H}\mathbf{P} - \mathbf{Y}\|_F^2 + \\ & \frac{\lambda_2}{2} \|\mathbf{H}\mathbf{C} - \mathbf{H}\|_F^2 + \frac{\lambda_3}{2} \text{tr}(\mathbf{W}\mathbf{L}\mathbf{W}^\top) + \lambda_4 \|\mathbf{W}\|_1 \\ \text{s.t.} \quad & \mathbf{H} \in [0, 1]^{n \times l}. \end{aligned} \tag{5}$$

For the problem (see the definition 1) addressed in this paper, we are also interested in what categories we have discovered and what their semantic concepts are. Motivated by previous studies (Huang et al. 2016, 2019; Wei et al. 2019; Wu et al. 2019; Zhang and Wu 2015) on learning label-specific features which have strong discrimination capabilities to each label, we add the ℓ_1 -norm regularization on the model coefficient matrix \mathbf{W} to learn the sparse label-specific features for each label, and expect to use them to describe the semantic meaning for the new discovered labels, and the results are provided in section 5.3.2.

It is noted that the formulation of the problem (5) is somewhat similar to the work SLEEC (Bhatia et al. 2016) on extreme multi-label classification. SLEEC aims to learn a low dimensional latent label space. While in our approach, we want to learn an augmented label space where the new discovered labels are paralleled with the observed labels, i.e., the new discovered labels have the same semantic level as the existing observed labels.

4 Optimization

For the problem (5), it is convex but non-smooth, and there are three coefficient parameters. We adopt the accelerated proximal gradient method (Beck and Teboulle 2009) to solve it, and update each parameter alternatively. We use $\mathcal{J}(\Psi)$ to represent the empirical loss of (5), where $\Psi = \{\mathbf{H}, \mathbf{W}, \mathbf{C}\}$ indicates the set of the three parameters.

4.1 Solving \mathbf{H}

By fixing \mathbf{W} and \mathbf{C} , the problem (5) becomes

$$\begin{aligned} \min_{\mathbf{H}} \quad & \frac{1}{2} \|\mathbf{XW} - \mathbf{HC}\|_F^2 + \frac{\lambda_0}{4} \|\mathbf{S} - \mathbf{HH}^\top\|_F^2 + \frac{\lambda_1}{2} \|\mathbf{HP} - \mathbf{Y}\|_F^2 \\ & + \frac{\lambda_2}{2} \|\mathbf{HC} - \mathbf{H}\|_F^2, \quad s.t. \mathbf{H} \in [0, 1]^{n \times l}. \end{aligned} \quad (6)$$

We can obtain the gradient w.r.t \mathbf{H} as

$$\begin{aligned} \nabla_{\mathbf{H}} \mathcal{J} = & (1 + \lambda_2)\mathbf{HCC}^\top - \mathbf{XWC}^\top + \lambda_0(\mathbf{HH}^\top - \mathbf{S})\mathbf{H} + \lambda_1(\mathbf{HP} - \mathbf{Y})\mathbf{P}^\top \\ & + \lambda_2(\mathbf{H} - \mathbf{H}(\mathbf{C} + \mathbf{C}^\top)). \end{aligned} \quad (7)$$

According the proximal gradient descend algorithm (Beck and Teboulle 2009), \mathbf{H} can be updated by

$$\mathbf{H} = \mathbf{H}^{(t)} - \frac{1}{L_f} \nabla_{\mathbf{H}} \mathcal{J}(\mathbf{H}^{(t)}, \mathbf{W}, \mathbf{C}), \quad (8)$$

where $\mathbf{H}^{(t)} = \mathbf{H}_t + \frac{\alpha_{t-1}-1}{\alpha_t}(\mathbf{H}_t - \mathbf{H}_{t-1})$. For a sequence α_t , it should satisfy the condition of $\alpha_t^2 - \alpha_t \leq \alpha_{t-1}^2$. Considering the non-negative constraint on $\mathbf{H} \in [0, 1]^{n \times m}$, \mathbf{H}

should be further post-processed by $\mathbf{H} = \max(\mathbf{H}, \mathbf{0})$ and the min-max normalization over each column of it. As a result, for each label, it has at least one positive example.

In Eq. (8), L_f indicates the Lipschitz constant. According to (Beck and Teboulle 2009), an approximate L_f can be obtained with a line-search strategy, where we keep updating $L_f = \eta L_f$, $\eta > 1$ until if it satisfies $\mathcal{J}(\Psi) < \mathcal{J}(\Psi') + \langle \nabla \mathcal{J}(\Psi'), \Psi - \Psi' \rangle + \frac{L_f}{2} \|\Psi - \Psi'\|_F^2$. Here, $\Psi' = \{\mathbf{H}^{(t)}, \mathbf{W}^{(t)}, \mathbf{C}^{(t)}\}$.

4.2 Solving \mathbf{W}

With \mathbf{H} and \mathbf{C} fixed, the problem (5) is simplified as

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{XW} - \mathbf{HC}\|_F^2 + \frac{\lambda_3}{2} \text{tr}(\mathbf{WLW}^\top). \quad (9)$$

Then, we can obtain the gradient w.r.t \mathbf{W} as

$$\nabla_{\mathbf{W}} \mathcal{J} = \mathbf{X}^\top (\mathbf{XW} - \mathbf{HC}) + \lambda_3 \mathbf{XWL}. \quad (10)$$

Consequently, \mathbf{W} can be updated by

$$\mathbf{W} = \mathbf{W}^{(t)} - \frac{1}{L_f} \nabla_{\mathbf{W}} \mathcal{J}(\mathbf{H}, \mathbf{W}^{(t)}, \mathbf{C}), \quad (11)$$

where $\mathbf{W}^{(t)} = \mathbf{W}_t + \frac{\alpha_t - 1}{\alpha_t} (\mathbf{W}_t - \mathbf{W}_{t-1})$. Considering the ℓ_1 -norm over parameter \mathbf{W} , the result can be further updated by the element-wise soft-threshold operator which is defined as

$$\mathbf{W} = \mathbf{prox}_{\frac{\lambda_4}{L_f}}(\mathbf{W}), \quad (12)$$

where $\mathbf{prox}_\epsilon(a)$ is the element-wise operator which is defined as

$$\mathbf{prox}_\epsilon(a) = \mathbf{sign}(a) \max(|a| - \epsilon, 0). \quad (13)$$

4.3 Solving \mathbf{C}

With \mathbf{H} and \mathbf{W} fixed, the problem (5) reduces to

$$\min_{\mathbf{C}} \frac{1}{2} \|\mathbf{XW} - \mathbf{HC}\|_F^2 + \frac{\lambda_2}{2} \|\mathbf{HC} - \mathbf{H}\|_F^2. \quad (14)$$

Then, we can obtain the gradient w.r.t \mathbf{C} as

$$\nabla_{\mathbf{C}} \mathcal{J} = \mathbf{H}^\top \mathbf{HC} - \mathbf{H}^\top \mathbf{XW} + \lambda_2 (\mathbf{H}^\top \mathbf{HC} - \mathbf{H}^\top \mathbf{H}). \quad (15)$$

Algorithm 1: Multi-Label Learning with Missing and Completely Unobserved Labels

Input: Training data matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$, label matrix $\mathbf{Y} \in \mathbb{R}^{n \times q}$, and weighting parameters $\lambda_0, \lambda_1, \lambda_2, \lambda_3, \lambda_4$; the number of unobserved labels r

Output: Model coefficient matrixes \mathbf{H}^* , \mathbf{W}^* , and \mathbf{C}^* .

- 1 **Initialization:** $b_0, b_1 \leftarrow 1$; $t \leftarrow 1$;
- 2 Initialize \mathbf{H} , \mathbf{W} and \mathbf{C} with random values;
- 3 **repeat**
- 4 compute L_f according to the line search strategy (Beck and Teboulle, 2009);
- 5 compute the graph Laplacian matrix \mathbf{L} ;
- 6 update \mathbf{H} by solving the sub-problem (6);
- 7 update \mathbf{W} by solving the sub-problem (9);
- 8 update \mathbf{C} by solving the sub-problem (14);
- 9 $b_t \leftarrow \frac{1 + \sqrt{4b_t^2 + 1}}{2}$;
- 10 $t \leftarrow t + 1$;
- 11 **until** stop criterion reached;
- 12 $\mathbf{H}^* \leftarrow \mathbf{H}_{t+1}$, $\mathbf{W}^* \leftarrow \mathbf{W}_{t+1}$, and $\mathbf{C}^* \leftarrow \mathbf{C}_{t+1}$;
- 13 return \mathbf{H}^* , \mathbf{W}^* , and \mathbf{C}^* .

Therefore, a closed-form solution to \mathbf{C} can be obtained as

$$\mathbf{C} = ((\lambda_2 + 1)\mathbf{H}^\top \mathbf{H})^{-1} (\mathbf{H}^\top \mathbf{X} \mathbf{W} + \lambda_2 \mathbf{H}^\top \mathbf{H}). \quad (16)$$

According to the above optimization procedures, we can summarize all the optimization steps of the proposed method in Algorithm 1.

5 Experiments

5.1 Experimental configuration

5.1.1 Dataset and configuration

The experiment is conducted over ten multi-label benchmark datasets, and the details of which are summarized in Table 1. For each data set, a 5-fold cross validation is adopted three times. To evaluate the performance on completely unobserved labels, we set the first $\lfloor 90\%l \rfloor$ labels as observed and the rest $\lceil 10\%l \rceil$ as unobserved labels, where l indicates the number of all the labels. In addition, to imitate missing labels, we randomly drop some of the labeling results of the $\lfloor 90\%l \rfloor$ observed labels for the training data of each dataset according to a predefined missing rate e.g., 10%, 15% and 20%.

5.1.2 Comparison approaches

By surveying previous studies on multi-label learning, it was found that there is no previous work on solving multi-label learning with missing and completely unobserved

Table 1 Description of datasets

Data set	#Instance	#Feature	#Label	Cardinality	Domain	URL
bibtex	7395	1836	159	2.40	text	URL1
corel16k001	13,766	500	153	2.86	image	URL1
corel16k002	13,761	500	164	2.88	image	URL1
corel16k003	13,760	500	154	2.83	image	URL1
medical	978	1449	45	1.25	text	URL1
stackex-chess	1675	585	227	2.41	text	URL2
stackex-cooking	10,491	577	400	2.23	text	URL2
stackex-cs	9270	635	274	2.56	text	URL2
stackex-philosophy	3971	842	233	2.27	text	URL2
stackex-chemistry	6961	540	175	2.11	text	URL2

URL1: <http://mulan.sourceforge.net/datasets-mlc.html>

URL2: <http://www.uco.es/kdis/mlresources/>

labels. To verify the effectiveness of our approach, we compare it with the following state-of-the-art multi-label classification approaches in terms of their performance on observed labels, and detailed configurations of them are summarized as below. The two approaches LSML (Huang et al. 2019) and Glocal (Zhu et al. 2018) can handle the problem of missing labels for multi-label learning. Parameter tuning for all the comparison approaches is based on a 5-fold cross validation over the training data of each dataset.

- BR (Boutell et al. 2004): Binary relevance. Ridge Regression is utilized as the base learner for each binary classifier of BR approach, and the regularization parameter is tuned in $\{10^i | i = -2, \dots, 2\}$.
- ECC (Read et al. 2009): Ensemble of classifier chains (CC). Ridge Regression is utilized as the base learner for each binary classifier of CC approach, and the regularization parameter is tuned in $\{10^i | i = -2, \dots, 2\}$. The ensemble size is set to be 15, and the chain order for each CC is generated randomly.
- MLkNN (Zhang and Zhou 2007):¹ A lazy learning approach to multi-label learning. The number of nearest neighbors k is tuned in $\{7, \dots, 17\}$.
- LSML (Huang et al. 2019):² It learns label-specific features for multi-label classification with missing labels, classification and label matrix recovery are performed jointly. All the parameters of it are searched in $\{10^i | i = -5, \dots, 3\}$.
- Glocal (Zhu et al. 2018):³ It can simultaneously recover the missing labels, train the linear classifiers, explore and exploit both global and local label correlations. Parameter $\lambda = 1$, λ_1 to λ_5 are searched in $\{10^i | i = -5, \dots, 1\}$, k is tuned in $\{0.1q, 0.2q, \dots, 0.6q\}$, and g is tuned in $\{5, 10, 15, 20\}$.
- MCUL: The proposed approach of this paper. MCUL-O is a simplified version of MCUL without discovering the unobserved labels, i.e., $k = 0$. Parameters λ_0 and

¹ code: <http://palm.seu.edu.cn/zhangml/files/ML-kNN.rar>.

² code: <https://jiunhwang.github.io/>.

³ code: <http://www.lamda.nju.edu.cn/MainPage.ashx>.

- λ_4 are tuned in $\{10^i | i = -1, \dots, 1\}$, λ_1 is tuned in $\{10^i | i = 0, \dots, 3\}$, λ_2 is tuned in $\{10^i | i = 0, \dots, 2\}$, and λ_3 is tuned in $\{5^i | i = 0, \dots, 3\}$.
- LSML-U and Glocal-U: Two different versions of LSML (Huang et al. 2019) and Glocal (Zhu et al. 2018) with a preprocessing step by solving the problem (2) to discover the unobserved labels for the training data. As a result, we can train LSML-U and Glocal-U on the full label matrix \mathbf{H} . It is worth noting that the algorithm Glocal-U needs an observation matrix to indicate which entities in the label matrix are observed (i.e., the value is not missing). Therefore, for Glocal-U, the entities of \mathbf{H} for the unknown labels are set as observed if the corresponding values are greater than 0.5.

5.2 Evaluation metrics

5.2.1 Evaluation metrics for observed labels

The performance of the comparison algorithms on observed labels is evaluated in terms of five common metrics (Gibaja and Ventura 2015; Herrera et al. 2016; Tsoumakas et al. 2010; Zhang and Zhou 2014), i.e., *One Error*, *Coverage*, *Ranking Loss*, *Average Precision* and *Macro AUC*.

5.2.2 Evaluation metrics for new discovered labels

To evaluate the performance on new discovered labels, we adopt F_U (Zhu et al. 2017) and propose three new metrics.

Given a test dataset, for the unobserved labels, $\bar{\mathbf{Y}} = [\bar{\mathbf{y}}_1, \bar{\mathbf{y}}_2, \dots, \bar{\mathbf{y}}_r] \in \{0, 1\}^{n_t \times r}$ indicates the ground truth of it, $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, \dots, \hat{\mathbf{y}}_r] \in \{0, 1\}^{n_t \times r}$ represents the predicted label matrix, and $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_r] \in \mathbb{R}^{n_t \times r}$ indicates the predicted score matrix.

- F_U was proposed in (Zhu et al. 2017). It measures the average label-based F_1 -measure on newly discovered and the ground-truth labels that best matches.

$$F_U = \frac{1}{r} \sum_{i=1}^r \max(\{F_1(\hat{\mathbf{y}}_i, \bar{\mathbf{y}}_j), j \in \{1, \dots, r\}\}) \quad (17)$$

where $F_1(\cdot)$ is the function to calculate the example-based F_1 score.

- RL_U measures the average label-based *Ranking Loss* on newly discovered and the ground-truth labels that best matches.

$$RL_U = \frac{1}{r} \sum_{i=1}^r \min(\{\text{RankingLoss}(\mathbf{a}_i, \bar{\mathbf{y}}_j), j \in \{1, \dots, r\}\}) \quad (18)$$

where $\text{Ranking Loss}(\cdot)$ evaluates the fraction of reversely ordered label pairs, i.e. an irrelevant label is ranked higher than a relevant label.

- Cov_U measures the average label-based *Coverage* on new discovered and the ground-truth labels that best matches.

$$\text{Cov}_U = \frac{1}{r} \sum_{i=1}^r \min(\{\text{Coverage}(\mathbf{a}_i, \bar{\mathbf{y}}_j), j \in \{1, \dots, r\}\}) \quad (19)$$

For a given output score \mathbf{a}_i , the function $\text{Coverage}(\cdot)$ evaluates how many steps are needed, on average, to move down the ranked label list so as to cover all the relevant labels of $\bar{\mathbf{y}}_j$. Consequently, the smaller the steps are, the better the performance is.

- LM_U measures the average *Label Matching* proportion over all the evaluation metrics.

$$\text{LM}_U = \frac{1}{m} \sum_{i=1}^m \frac{|\mathcal{S}_i \wedge \bar{\mathcal{Y}}|}{r} \quad (20)$$

where m is the number of metrics which can return a set of matched labels, and \mathcal{S}_i indicates the set of matched labels returned by the i -th metric. This metric indicates the average proportion of ground-truth labels that we have discovered among the new discovered labels.

For F_U and LM_U , the bigger the values of them are, the better the performance is. While for RL_U and Cov_U , the smaller the values of them are, the better the performance is.

5.3 Experiment results

As the compared approaches cannot solve multi-label learning with missing and completely unobserved directly, we evaluate the performance of them on observed and completely unobserved labels respectively.

5.3.1 Results on observed labels

The experimental results of each comparison algorithm on the observed labels are shown in Fig. 3. Moreover, we calculate the average results of each comparison approach over the ten data sets in terms of different evaluation metrics under different missing rates respectively, and the results are shown in Fig. 5, where the symbol \uparrow (\downarrow) indicates the larger (smaller) the value is, the better the performance is.

To analyze the relative performance among the comparison algorithms systematically, Friedman test (Demšar 2006) is employed to conduct performance analysis. The missing rate of observed labels is varied in the range of {10%, 15% 20%}, and as a result, there are 30 (3×10) data points totally. Table 2 summarizes the Friedman statistics F_F and the corresponding critical value in terms of each evaluation metric. As shown in Table 2, at significance level $\alpha = 0.05$, the null hypothesis that all the comparison algorithms perform equivalently is clearly rejected in terms of each evaluation metric. Consequently, we employ the Nemenyi test (Demšar 2006) to test whether our

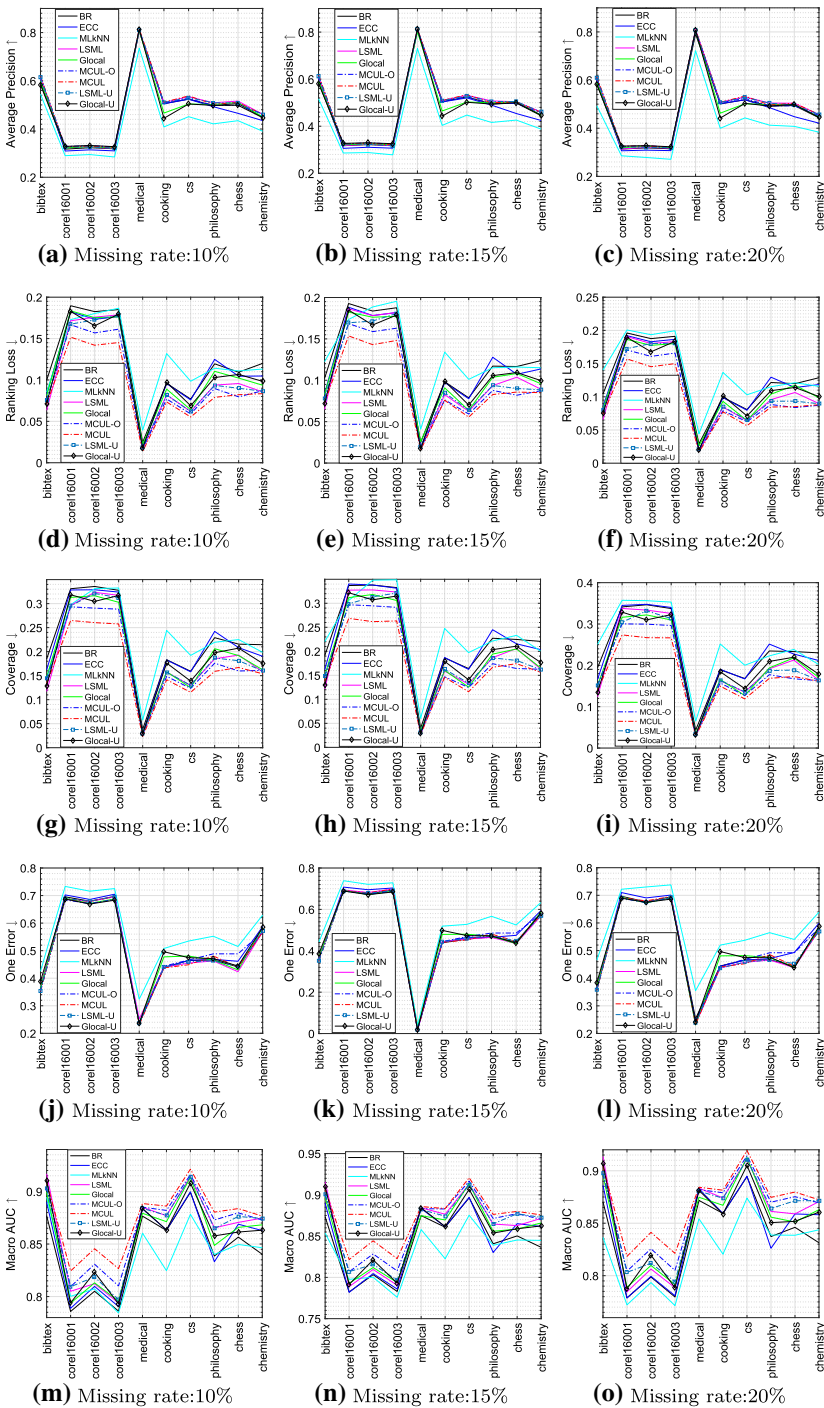


Fig. 3 Results of each comparison approach over the ten data sets in terms of all the evaluation metrics

Table 2 Summary of the Friedman Statistics F_F ($k = 9$, $N = 30$) and the critical value in terms of each evaluation metric (k : # comparison algorithms; N : # data points)

Metric	F_F	Critical value ($\alpha = 0.05$)
Average Precision	44.5073	1.9785
One Error	17.5268	
Ranking Loss	98.4622	
Coverage	107.9839	
AUC	104.4299	

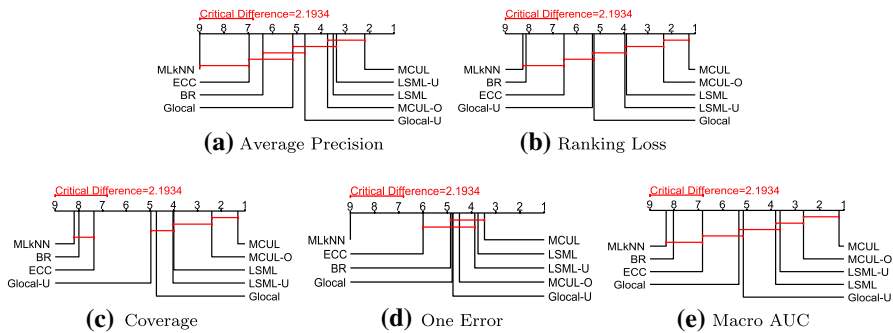


Fig. 4 Comparison of MCUL against the comparison approaches with the Nemenyi test. Groups of classifiers that are not significantly different from MCUL (at $p = 0.05$) are connected

proposed method MCUL achieves a competitive performance against the comparison algorithms, where MCUL is considered as the control algorithm. The performance between two classifiers will be significantly different if their average ranks differ by at least one critical difference $CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$. For Nemenyi test, $q_\alpha = 2.948$ at significance level $\alpha = 0.05$, and thus $CD = 2.1934$ ($k = 9$, $N = 30$). Fig. 4 shows the CD diagrams on each evaluation metric. In each sub-figure of Fig. 4, any comparison algorithm whose average rank is within one CD is connected. Otherwise, any algorithm not connected is considered to have significantly different performance between them. According to these experimental results, the following observations can be made:

- As shown in Fig. 5, the performance of each approach decreases with the increasing of missing rate. It verifies the importance of solving the problem of missing labels for multi-label learning.
- The proposed method MCUL significantly outperforms all the comparison approaches in terms of *Ranking Loss*, *Coverage*, and *Macro AUC*, and achieves statistically superior performance to other comparison approaches in terms of *Average Precision* and *One Error*. The superiority implies the effectiveness of the proposed method on multi-label learning with missing labels.
- The proposed method MCUL achieves statistically superior or at least comparable performance against its simplified version MCUL-O in terms of all the evaluation metrics. The superior performance of MCUL against MCUL-O definitely verifies

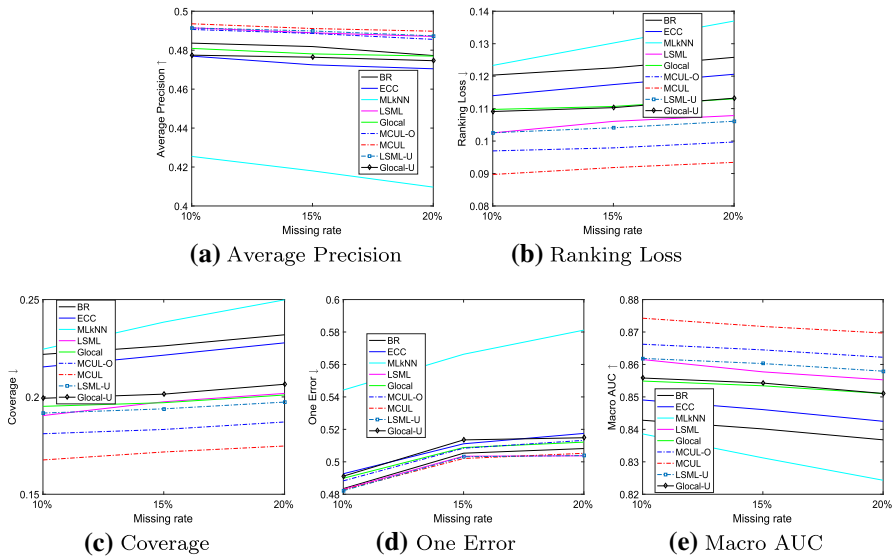


Fig. 5 Average results of each comparison approach over the ten data sets in terms of all the evaluation metrics under different missing rates of observed labels

that discovering and modeling the unobserved labels can improve the performance of our method on existing observed labels.

- LSML-U and GLocal-U outperform their original versions respectively in terms of *Average Precision* and *Macro AUC*, and achieve comparable performance against their original versions in terms of other evaluation metrics. This observation also verifies that discovering and modeling the unobserved labels can improve the performance on existing observed labels.
- MCUL achieves statistically superior performance to LSML and GLocal and their two extended versions in terms of all the evaluation metrics. The superior performance of MCUL demonstrates that our method can handle missing labels better than them.
- MLkNN achieves the worst performance on all the data sets. It is worth noting that MLkNN is constructed based on the information of k nearest neighbors of each instance. When the data set is with *missing labels*, especially some of the labels are *completely unobserved*, instances of the k nearest neighbors cannot provide sufficient information for MLkNN to learn reliable prior and posterior probabilities for the prediction. It implies the importance of solving data set with *missing and completely unobserved labels*.

5.3.2 Results on unobserved labels

In this section, we provide both the quantitative and qualitative analysis of the results on unobserved labels.

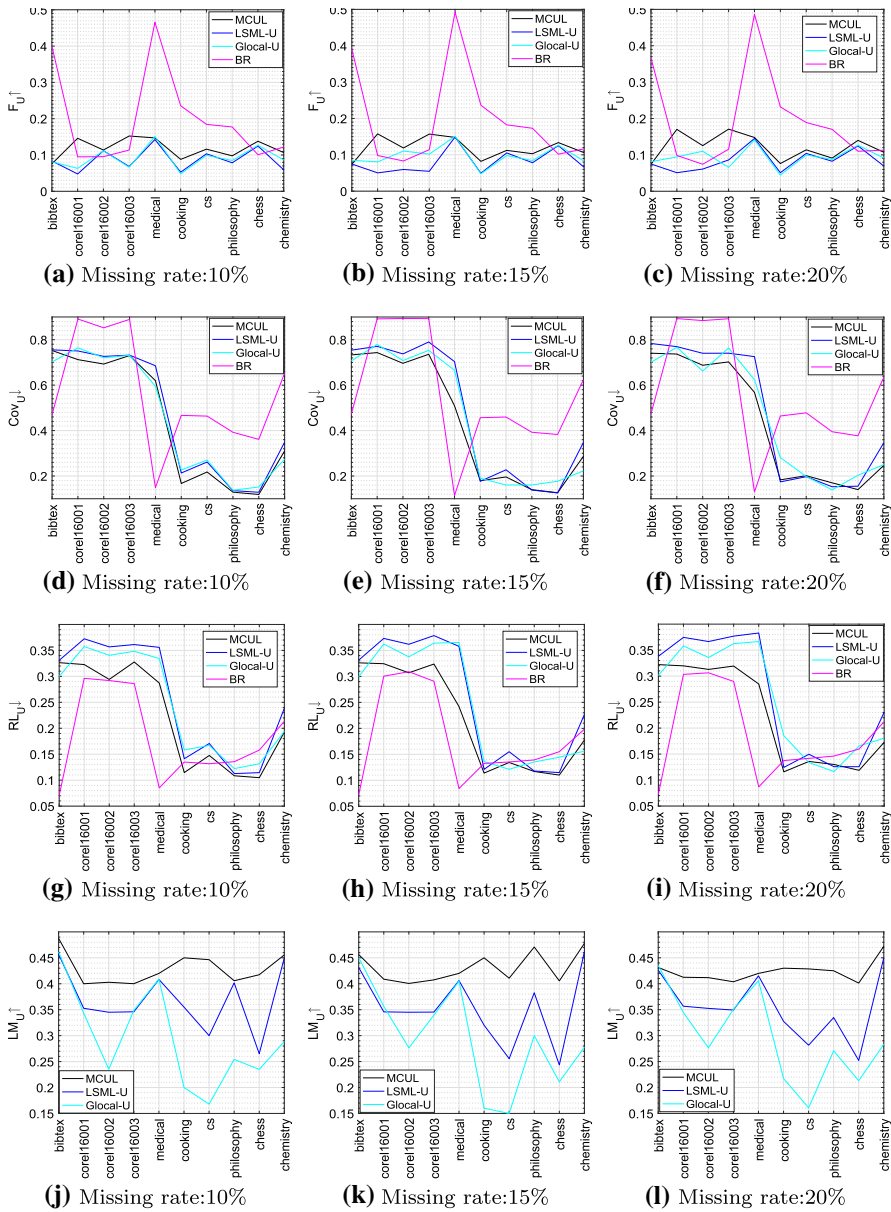


Fig. 6 Experimental results on the *Unobserved labels*. BR is trained based on the ground-truth of the unobserved labels

For the quantitative analysis, we compared MCUL with LSML-U, Glocal-U (*For detailed settings, please refer to Sect. 5.1.2*), and BR. Since BR cannot discover unobserved labels, we trained it based on the ground-truth of the unobserved labels, i.e., $h : \mathcal{X} \rightarrow 2^{\bar{\mathcal{Y}}}$. Although this comparison is unfair, we can still make some observations according to the results. For MCUL, LSML-U, and Glocal-U, $\bar{\mathcal{Y}}$ is unavailable during the training stage, the prediction threshold of them is tuned according to the result of example-based F_1 measure on observed labels respectively. Figure 6 shows the results of them on discovered unobserved labels in terms of F_U , RL_U , Cov_U , and LM_U (*Detailed definitions are provided in Sect. 5.2.2*). According to these results, we have the following observations:

- First, on most of the data sets, MCUL achieves a better performance than LSML-U and Glocal-U in terms of the four evaluation metrics. One possible reason might be that some of the labelling results of the unobserved labels for the training data by a preprocessing step are incorrect, and these incorrect results cannot be well optimized during the training stage of LSML-U and Glocal-U respectively.
- Second, MCUL achieves a better performance than BR in 40% cases in terms of F_U and RL_U . Besides, MCUL achieves a better performance than BR in 80% cases in terms of Cov_U . These results indicate that MCUL can discover some of the unobserved labels and the results of them are acceptable.
- Last, the results of LM_U of MCUL are located in the range of [40, 49]. These results clearly indicate that the proposed method can discover at least 40% of the unobserved labels. In addition, according to Fig. 7f, MCUL will achieve a better performance on the unobserved labels if we set a smaller value of r , i.e., the number of unobserved labels. For example, the value of LM_U can reach 85.6% for *stackex-cs* dataset when $r = 5$.

For the qualitative analysis, we want to show what categories we have discovered and what their semantic concepts are. As discussed in Sect. 3.3, we try to learn the sparse label-specific features for each label, and expect to use them to describe the semantic meaning of new discovered labels. In this section, we provide results of top 20 features and F_U of the five best matched labels (the criterion F_U is adopted) for three datasets, e.g., *stackex-cooking*, *stackex-cs*, and *stackex-philosophy*. For each dataset, the first $\lfloor 90\%l \rfloor$ labels are set as the observed labels and the missing rate is 10%, and the rest $\lfloor 10\%l \rfloor$ labels are set as unobserved labels, where l indicates the total number of labels.

The name of top-five best matched ground-truth labels and their matched results F_U , and top 20 features of each dataset are shown in Tables 3, 4 and 5, where the features are arranged in a descending order according to the values of \mathbf{W} . Specifically, if the i -th newly discovered label is best matched with the j -th ground-truth label according to $F_U(i, j)$, then y_j is the name of best matched ground-truth label, and the top 20 features are arranged in a descending order by sorting the values of $|\mathbf{w}^{(q+i)}|$. As shown in the Tables 3, 4 and 5, for each matched ground truth label, in most cases, the name of it ranks in the first or second place among the corresponding top 20 label-specific features. In addition, most of these features have a strong semantic correlation with it.

In Table 5, it is noted that the labels *Theology* and *Stoicism* do not appear in the top 20 features. We also find that the word *Stoicism* does not exist in the feature space.

Table 3 Top 20 features for the five best matched labels of *stackex-cooking*

Label name	F_U	Top 20 features (i.e., feature name (weight))
Tomatoes	0.602	tomato (0.504), supermarket(0.118), recent(0.110), meal(0.106), includ(0.106), plastic(0.106), cool(0.101), told(0.100), process(0.098), theyr(0.098), ill(0.097), stop(0.096), post(0.096), tonight(0.095), bacon(0.094), past(0.093), reduc(0.092), lose(0.088), stuff(0.088), skin(0.088)
Yeast	0.333	yeast (0.416), packag(0.121), microwav(0.110), mention(0.106), fact(0.104), want(0.103), produc(0.098), onlin(0.097), ideal(0.096), let(0.094), figur(0.094), techniqu(0.093), raw(0.091), thaw(0.091), possibl(0.089), life(0.086), left(0.086), cooki(0.085), solid(0.084), remov(0.083)
Wine	0.315	wine (0.414), part(0.139), box(0.135), peopl(0.125), addit(0.116), pretti(0.106), place(0.103), acid(0.103), book(0.102), simpl(0.102), step(0.098), great(0.098), past(0.097), morn(0.095), term(0.094), find(0.094), result(0.093), mix(0.091), smell(0.091), medium(0.090)
Sugar	0.301	sugar (0.481), pork(0.138), quick(0.126), true(0.121), duplic(0.114), sticki(0.109), simmer(0.105), pound(0.101), purchas(0.101), import(0.100), expens(0.100), worri(0.099), salad(0.098), knife(0.098), alcohol(0.098), butter(0.098), fruit(0.094), juic(0.090), previous(0.089), depend(0.087)
Vinegar	0.232	vinegar (0.437), turkey(0.129), orang(0.124), amount(0.120), groceri(0.113), jar(0.112), creat(0.111), smell(0.104), frozen(0.103), feel(0.101), batch(0.100), find(0.096), purchas(0.096), worri(0.094), case(0.094), ground(0.093), number(0.092), state(0.091), color(0.090), knife(0.090)

Besides, we extracted a brief introduction to these two topics *Theology*⁴ and *Stoicism*⁵ from Wikipedia respectively. It is found that the top 20 features of these two labels still have strong semantic correlations with the name of labels.

Therefore, we argue that the semantic meaning of the discovered labels can be depicted by these label-specific features. For image data, if the features are extracted based on the sub-area of images or high level features learned by some advanced approaches, such as deep learning approaches, we think that this strategy can also work well. Moreover, if we have the raw data, we can better describe the semantic meaning. The proposed method MCUL can predict both the observed and unobserved

⁴ Theology is the systematic study of the nature of the divine and, more broadly, of religious belief. It is taught as an academic discipline, typically in universities and seminaries. It occupies itself with the unique content of analyzing the supernatural, but also deals with religious epistemology, asks and seeks to answer the question of revelation. Revelation pertains to the acceptance of **God**, gods, or deities, as not only transcendent or above the natural world, but also willing and able to interact with the natural world and, in particular, to reveal themselves to humankind. While theology has turned into a secular field, religious adherents still consider theology to be a discipline that helps them live and understand concepts such as life and love and that helps them lead lives of obedience to the deities they follow or worship.

⁵ Stoicism is a philosophy of personal ethics informed by its **system** of logic and its views on the natural world. According to its teachings, as **social** beings, the path to eudaimonia (happiness, or blessedness) for humans is found in accepting the moment as it presents itself, by not allowing oneself to be controlled by the desire for pleasure or fear of pain, by using one's mind to understand the world and to do one's part in nature's plan, and by working together and treating others fairly and justly.

Table 4 Top 20 features for the five best matched labels of *stackex-cs*

Label name	F_U	Top 20 features (i.e., feature name (weight))
Turing-machines	0.484	ture (0.412), machin (0.366), tape(0.163), link(0.106), paramet(0.106), count(0.103), wonder(0.101), cover(0.100), repeat.(0.098), formul(0.096), infinit(0.094), author(0.093), let(0.093), correspond(0.092), heurist(0.088), general(0.087), reason(0.087), locat(0.087), simul(0.086), seri(0.086)
Trees	0.279	tree (0.533), figur(0.146), binari(0.133), vertex(0.126), refer(0.121), topic(0.113), insert(0.099), level(0.098), sourc(0.097), point(0.097), exam(0.096), reason(0.096), perform(0.095), belong(0.094), store(0.094), constant(0.092), pars(0.091), compar(0.091), contradict(0.090), suppos(0.090)
Strings	0.211	string (0.473), follow(0.131), natur(0.131), reason(0.104), text(0.102), reduc(0.097), unit(0.096), semant(0.091), contradict(0.091), mathemat(0.089), runtim(0.089), amount(0.088), statement(0.088), recurs(0.088), bound(0.088), neq(0.088), optim(0.087), singl(0.086), answer(0.085), group(0.085)
Undecidability	0.204	decid (0.354), undecid (0.148), verifi(0.134), degre(0.123), select(0.121), file(0.119), make(0.108), determinist(0.107), deriv(0.107), import(0.107), maximum(0.100), sort(0.098), mathemat(0.098), normal(0.096), definit(0.094), fit(0.094), wonder(0.088), produc(0.085), professor(0.084), index(0.082)
Time-complexity	0.189	time (0.536), complex (0.165), wrong(0.125), thought(0.125), hard(0.118), subseteq(0.115), final(0.114), memori(0.113), decis(0.102), iter(0.101), row(0.101), undecid(0.100), connect(0.099), number(0.098), polynomi(0.097), undirect(0.096), prove(0.095), belong(0.094), peopl(0.094), simpl(0.091)

labels for data examples. After the prediction, we will know the tentative labelling results of the data examples.

5.3.3 Parameter analysis

The average results (i.e., *Average Precision* and *One Error*) of MCUL with different values of λ_0 , λ_1 , λ_2 , λ_3 , and λ_4 over *stackex-cs* are shown in Fig. 7a–e. It is noted that the performance of MCUL is insensitive to the parameters, and also the optimal performance is usually achieved at some intermediate values of the parameters.

Figure 7f shows the average results of MCUL over 15 repetitions with different numbers of unobserved labels. The result (i.e., *One Error*) on observed labels is improved and then dropped down with the increasing of the number of unobserved class labels (i.e., r), and MCUL obtains the best performance when $r = \lceil 10\%l \rceil = 28$, where $l = 274$ for *stackex-cs*. As shown in Fig. 7f, the result (i.e., F_U) on unobserved labels decreases with the growing of the number of unobserved class labels. Therefore, it is reasonable that the difficulty of discovering the unobserved labels increases with the growing of the number of unobserved labels, i.e. the larger the number of unobserved class labels is, the more difficult it is to discover them. Therefore, we think that an appropriate value of r could be searched by cross-validation according to the performance of observed labels. This strategy is feasible for data set with a small number of

Table 5 Top 20 features for the five best matched labels of *stackex-philosophy*

Label name	F_U	Top 20 features (i.e., feature name (weight))
Theology	0.366	god (0.600), free(0.117), anim(0.115), kind(0.113), subject(0.111), modern(0.107), die(0.106), choic(0.102), thought(0.101), fact(0.101), essenc(0.100), imposs(0.097), space(0.094), sentenc(0.093), peopl(0.091), product(0.090), natur(0.087), term(0.086), justifi(0.085), quot(0.085)
Truth	0.132	statement(0.454), truth (0.349), true(0.320), fals(0.188), interest(0.139), histori(0.118), belief(0.117), clear(0.112), interact(0.105), comput(0.105), contradict(0.100), give(0.096), thought(0.095), proof(0.092), ontolog(0.091), simpli(0.090), paper(0.089), sentenc(0.088), relev(0.085), thing(0.085)
Time	0.111	time (0.623), life(0.134), materi(0.133), intuit(0.132), assum(0.130), event(0.116), earth(0.110), posit(0.107), god(0.104), start(0.101), energi(0.099), brain(0.094), proof(0.093), edit(0.092), happi(0.092), begin(0.092), equal(0.089), properti(0.088), realism(0.086), nietsch(0.086)
Stoicism	0.108	govern(0.216), social(0.163), explain(0.121), talk(0.107), class(0.103), proof(0.101), system(0.100), intuit(0.100), consist(0.099), major(0.097), classic(0.088), contemporari(0.088), right(0.087), truth(0.087), part(0.086), world(0.086), religi(0.085), let(0.083), design(0.082), peopl(0.080)
Wittgenstein	0.098	philosoph(0.534), thought(0.135), fals(0.127), ration(0.118), realiti(0.116), life(0.115), specif(0.111), religion(0.109), law(0.106), process(0.106), wittgenstein (0.102), form(0.101), theori(0.100), issu(0.099), possibl(0.099), axiom(0.098), correct(0.095), ontolog(0.093), talk(0.091), believ(0.090)

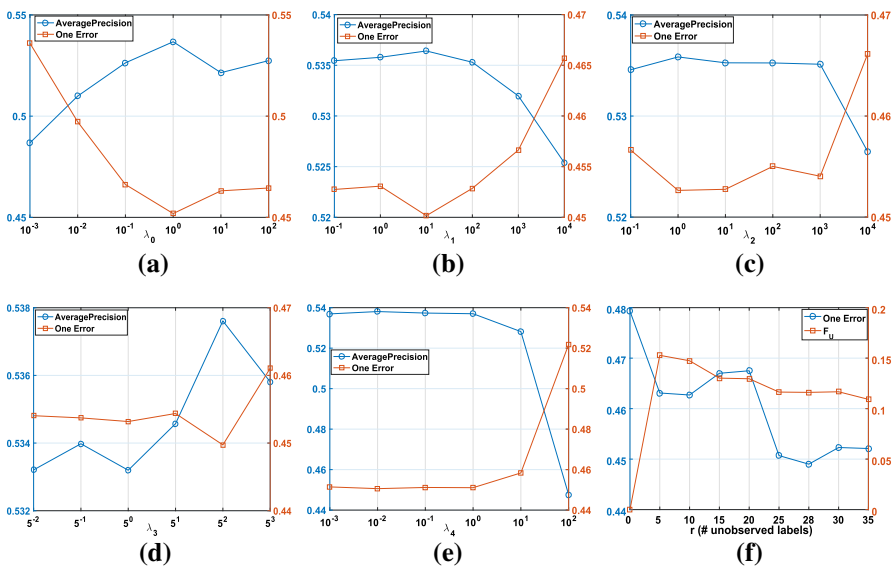


Fig. 7 Parameter analysis on MCUL over *stackex-cs*

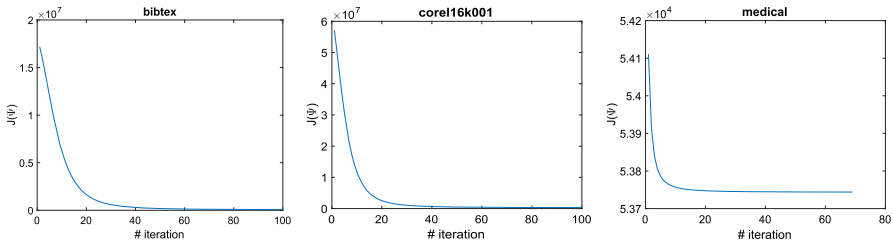


Fig. 8 Examples of the convergence curve of MCUL

class labels. For a data set with extreme number of class labels, the parameter range for r will be too large, and here we provide a possible way to run our model in real application. Specifically, we can set a small step r_t for the number of unobserved labels, and run the model multiple times until the performance on the observed labels becomes worse or unacceptable.

5.3.4 Complexity and convergence

For the proposed objective function (5), the most time-consuming step is to calculate the second term $\|\mathbf{S} - \mathbf{H}\mathbf{H}^T\|_F^2$ which leads to a time complexity of $\mathcal{O}(n^2(d + l + n))$ and a memory complexity of $\mathcal{O}(n^2)$, where n indicates the number of data instances, d and l represent the number of features and labels respectively.

Figure 8 shows the value $\mathcal{J}(\Psi)$ of the objective function (5) of MCUL *w.r.t* the number of iterations over *bibtex*, *corel16k001* and *medical* three datasets, respectively. It is noted that the value of $\mathcal{J}(\Psi)$ drops sharply around 30 iterations and then tends to become stable. For the other experimental data sets, the proposed method MCUL can converge within 100 iterations at most.

6 Conclusion

In this paper, we propose a new approach named MCUL to solve multi-label learning with missing and completely unobserved labels. It can not only discover the unobserved labels for the training data but also predict new data examples with the observed and new discovered labels simultaneously. The experimental results demonstrate the effectiveness of our method, and verify the importance of discovering and modeling unobserved label for multi-label learning.

This work tentatively solves the problem of multi-label learning with missing and completely unobserved labels. We think that this problem may have a long-term benefit to the community of multi-label learning. There are a few issues to this method that will be considered in a future study. First, how to automatically decide the number of the unobserved labels. In real applications, there is no prior knowledge about it. Second, how to describe the semantic meaning of the unobserved labels for various types of data. Third, the proposed problem can be solved together with many other

challenging problems in multi-label learning, such as online learning, semi-supervised multi-label learning, feature selection, extreme multi-label learning, etc.

Acknowledgements This work is supported by NSFC: 61806005, JST KAKENHI: 191400000190, and JST-AIP: JPMJCR19U4.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Beck A, Teboulle M (2009) A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J Imaging Sci* 2(1):183–202
- Bhatia K, Jain H, Kar P, Varma M, Jain P (2016) Sparse local embeddings for extreme multi-label classification. In: *Neural information processing systems (NIPS)*, pp 730–738
- Boutell MR, Luo JB, Shen XP, Brown CM (2004) Learning multi-label scene classification. *Pattern Recognit* 37(9):1757–1771
- Chu HM, Huang KH, Lin HT (2019) Dynamic principal projection for cost-sensitive online multi-label classification. *Mach Learn* 108(1):1193–1230
- Da Q, Yu Y, Zhou ZH (2014) Learning with augmented class by exploiting unlabeled data. In: *AAAI conference on artificial intelligence (AAAI)*, pp 2373–2379
- Decubber S, Mortier T, Dembczyński K, Waegeman W (2019) Deep f-measure maximization in multi-label classification: A comparative study. In: *European conference on machine learning and principles and practice of knowledge discovery in databases (ECML PKDD)*, pp 290–305
- Dembczyński K, Cheng W, Hüllermeier E (2010) Bayes optimal multilabel classification via probabilistic classifier chains. In: *International conference on machine learning (ICML)*, pp 1609–1614
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7:1–30
- Ding C, He X, Simon HD (2005) On the equivalence of nonnegative matrix factorization and spectral clustering. In: *SIAM international conference on data mining (SDM)*, pp 606–610
- Elisseeff A, El Ghemal W (2001) A kernel method for multi-labelled classification. In: *Neural information processing systems (NIPS)*, pp 681–687
- Fürnkranz J, Hüllermeier E, Loza Mencía E, Brinker K (2008) Multilabel classification via calibrated label ranking. *Mach Learn* 73(2):133–153
- Gibaja E, Ventura S (2015) A tutorial on multilabel learning. *ACM Comput Surv* 47(3):52:1–52:38
- Herrera F, Charte F, Rivera AJ, del Jesus MJ (2016) *Multilabel classification: problem analysis, metrics and techniques*. Springer, Berlin
- Huang J, Li G, Huang Q, Wu X (2016) Learning label-specific features and class-dependent labels for multi-label classification. *IEEE Trans Knowl Data Eng* 28(12):3309–3323
- Huang J, Li G, Huang Q, Wu X (2018) Joint feature selection and classification for multilabel learning. *IEEE Trans Cybern* 48(3):876–889
- Huang J, Qin F, Zheng X, Cheng Z, Yuan Z, Zhang W, Huang Q (2019) Improving multi-label classification with missing labels by learning label-specific features. *Inf Sci* 492:124–146
- Liu W (2019) Copula multi-label learning. In: *Neural information processing systems (NIPS)*, pp 6334–6343
- Liu W, Shen X (2019) Sparse extreme multi-label learning with oracle property. In: *International conference on machine learning (ICML)*, pp 4032–4041
- Masera L, Blanzieri E (2019) Awx: an integrated approach to hierarchical-multilabel classification. In: *European conference on machine learning and principles and practice of knowledge discovery in databases (ECML PKDD)*, pp 322–336

- Mu X, Zhu F, Du J, Lim EP, Zhou ZH (2017) Streaming classification with emerging new class by class matrix sketching. In: AAAI conference on artificial intelligence (AAAI), pp 2373–2379
- Nguyen V, Hüllermeier E (2019) Reliable multi-label classification: prediction with partial abstention. CoRR arXiv:1904.09235
- Park LAF, Read J (2019) A blended metric for multi-label optimisation and evaluation. In: European conference on machine learning and principles and practice of knowledge discovery in databases (ECML PKDD), pp 719–734
- Pham A, Raich R, Fern X, Arriaga JP (2015) Multi-instance multi-label learning in the presence of novel class instances. In: International conference on machine learning (ICML), pp 2427–2435
- Qu W, Zhang Y, Zhu J, Qiu Q (2009) Mining multi-label concept-drifting data streams using dynamic classifier ensemble. In: Asian conference on machine learning (ACML), pp 308–321
- Read J, Pfahringer B, Holmes G (2008) Multi-label classification using ensembles of pruned sets. In: IEEE international conference on data mining (ICDM), pp 995–1000
- Read J, Pfahringer B, Holmes G, Frank E (2009) Classifier chains for multi-label classification. In: European conference on machine learning and principles and practice of knowledge discovery in databases (ECML PKDD), pp 254–269
- Read J, Bifet A, Holmes G, Pfahringer B (2011) Streaming multi-label classification. In: Proceedings of 2nd workshop applications of pattern analysis, vol 17, pp 19–25
- Shi Z, Xue Y, Wen Y, Cai G (2014) Efficient class incremental learning for multi-label classification of evolving data streams. In: International joint conference on neural network (IJCNN), pp 2093–2099
- Sun Y, Zhang Y, Zhou Z (2010) Multi-label learning with weak label. In: AAAI conference on artificial intelligence (AAAI), pp 593–598
- Tan Q, Yu G, Domeniconi C, Wang J, Zhang Z (2018) Incomplete multi-view weak-label learning. In: International joint conference on artificial intelligence (IJCAI), pp 2703–2709
- Tsoumakas G, Katakis I, Vlahavas I (2010) Mining multi-label data. In: Data mining and knowledge discovery handbook, pp 667–685
- Tsoumakas G, Katakis I, Vlahavas L (2011) Random k-labelsets for multilabel classification. IEEE Trans Knowl Data Eng 23(7):1079–1089
- Wei T, Tu W, Li Y (2019) Learning for tail label data: a label-specific feature approach. In: International joint conference on artificial intelligence (IJCAI), pp 3842–3848
- Wu X, Chen Q, Hu Y, Wang D, Chang X, Wang X, Zhang ML (2019) Multi-view multi-label learning with view-specific information extraction. In: International joint conference on artificial intelligence (IJCAI), pp 3884–3890
- Wydmuch M, Jasinska K, Kuznetsov M, Busa-Fekete R, Dembczynski K (2018) A no-regret generalization of hierarchical softmax to extreme multi-label classification. In: Neural information processing systems (NIPS), pp 6355–6366
- Xu M, Jin R, Zhou Z (2013) Speedup matrix completion with side information: application to multi-label learning. In: Neural information processing systems (NIPS), pp 2301–2309
- Yu H, Jain P, Kar P, Dhillon IS (2014) Large-scale multi-label learning with missing labels. In: International conference on machine learning (ICML), pp 593–601
- Zhang J, Wu X (2019) Multi-label truth inference for crowdsourcing using mixture models. IEEE Trans Knowl Data Eng. <https://doi.org/10.1109/TKDE.2019.2951668>
- Zhang ML, Wu L (2015) Lift: Multi-label learning with label-specific features. IEEE Trans Pattern Anal Mach Intell 37(1):107–120
- Zhang ML, Zhou ZH (2006) Multilabel neural networks with applications to functional genomics and text categorization. IEEE Trans Knowl Data Eng 18(10):1338–1351
- Zhang ML, Zhou ZH (2007) MI-knn: a lazy learning approach to multi-label learning. Pattern Recognit 40(7):2038–2048
- Zhang ML, Zhou ZH (2014) A review on multi-label learning algorithms. IEEE Trans Knowl Data Eng 26(8):1819–1837
- Zhang Y, Heno R, Gan Z, Li Y, Carin L (2018) Multi-label learning from medical plain text with convolutional residual models. In: Proceedings of the 3rd machine learning for healthcare conference, pp 280–294
- Zhang YJ, Zhao P, Zhou ZH (2020) Exploratory machine learning with unknown unknowns. CoRR arXiv:2002.01605
- Zhu Y, Ting KM, Zhou ZH (2017) Discover multiple novel labels in multi-instance multi-label learning. In: AAAI conference on artificial intelligence (AAAI), pp 2977–2983

- Zhu Y, Kwok JT, Zhou ZH (2018) Multi-label learning with global and local label correlation. *IEEE Trans Knowl Data Eng* 30(6):1081–1094
- Zhu Y, Ting KM, Zhou ZH (2018) Multi-label learning with emerging new labels. *IEEE Trans Knowl Data Eng* 30(10):1901–1914

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

Authors and Affiliations

Jun Huang¹  · Linchuan Xu² · Kun Qian³ · Jing Wang⁴ · Kenji Yamanishi⁵

✉ Jun Huang
huangjun.cs@ahut.edu.cn

- ¹ School of Computer Science and Technology, Anhui University of Technology, Ma'anshan, China
- ² Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong SAR, China
- ³ Educational Physiology Laboratory, The University of Tokyo, Tokyo, Japan
- ⁴ School of Computing and Mathematical Sciences, University of Greenwich, Greenwich, UK
- ⁵ Graduate School of Information Science and Technology, The University of Tokyo, Tokyo, Japan