SCIENCE CHINA
Information Sciences

March 2021, Vol. 64 130101:1–130101:13
https://doi.org/10.1007/s11432-020-3132-4

• RESEARCH PAPER •
Special Focus on Weakly Supervised Learning

Learning from group supervision: the impact of supervision deficiency on multi-label learning

Miao XU[1,2*] & Lan-Zhe GUO[3*]

[1]*School of Information Technology and Electrical Engineering, The University of Queensland, St Lucia* 4072*, Australia;*
[2]*RIKEN Center for Advanced Intelligence Project, Tokyo* 103-0027*, Japan;*
[3]*National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing* 210023*, China*

Received 30 June 2020/Accepted 30 July 2020/Published online 7 February 2021

Abstract Multi-label learning studies the problem where one instance is associated with multiple labels. Weakly supervised multi-label learning has attracted considerable research attention because of the annotation difficulty. Majority of the studies on weakly supervised multi-label learning assume that one group of weak annotations is available for each instance; however, none of these studies considers multiple groups of weak annotations that can be easily acquired through crowdsourcing. Recent studies on crowdsourced multi-label learning observed that the current query strategies do not agree well with human habits and that data cannot be collected as expected. Therefore, this study aims to design a new query strategy in accordance with human behavior patterns to obtain multiple groups of weak annotations. Further, a learning algorithm is proposed based on neural networks for such type of data. In addition, this study qualitatively and empirically analyzes factors in the proposed query strategy that may impact further learning and provides insights to obtain better query strategy with respect to future crowdsourcing in case of multi-label data.

Keywords multi-label learning, weakly supervised leaning, weakly supervised multi-label learning, query strategy, group supervision

Citation Xu M, Guo L-Z. Learning from group supervision: the impact of supervision deficiency on multi-label learning. Sci China Inf Sci, 2021, 64(3): 130101, https://doi.org/10.1007/s11432-020-3132-4

1 Introduction

Multi-label learning [1], which assumes one instance is associated with multiple labels simultaneously, has been studied for decades and achieved successful application in various domains, including image classification [2,3], text classification [4,5] and computational biology [6,7].

Despite the success, one of the problems associated with multi-label learning is that the annotation information is difficult to obtain, resulting in a weakly supervised problem. The weakly supervised problem associated with general machine learning has been investigated in [8], in which such type of problem is categorized into three different types: incomplete supervision, inexact supervision and inaccurate supervision. The weakly supervised problem associated with multi-label learning can also be categorized into these three types. In [9], only some of the relevant and irrelevant labels are assumed to be annotated. These types of studies can be considered as multi-label learning with "incomplete supervision". In some studies, weak labels have been considered wherein only some of the relevant labels are available [10,11]. These studies can be referred to as multi-label learning with "inexact supervision". Recently, multi-label learning with partial labels [12,13] has become popular, wherein a candidate label set containing all true labels is given for instance. These studies can be considered as multi-label learning with "inaccurate supervision".

These studies on weakly supervised multi-label learning assume that only one source of supervision is provided for an instance. In reality, crowdsourcing [14] is always used. Thus, we can collect data from crowds to achieve data annotation, resulting in different sources of annotations. The existing studies on crowdsourced multi-label learning [15,16] assume that each user is given all labels and that users annotate

* Corresponding author (email: miao.xu@uq.edu.au, guolz@lamda.nju.edu.cn)

each instance label by label based on their expertise. Finally, each label-instance pair is expected to have an annotation. Ref. [15] observed that users tend to annotate only a few relevant labels before they give up by studying the real crowd annotated multi-label data. Hence, current query strategies for users to annotate in a label-by-label manner may not be in agreement with the human behavior pattern; thus, the annotation task cannot be completed in the expected way. Although Ref. [15] tackled this problem by estimating the expertise of the users, the main problem is whether we can resolve this problem from another viewpoint, i.e., by designing a better annotation strategy that agrees well with the behavior pattern of humans, thereby reducing the heavy load of the annotation task.

To achieve this, we must first understand human behavior patterns. Neuroscientists have studied the manner in which humans look at an image through eye tracking [17]. They observed that there are variations in the order which parts of the image are gazed at first, and which parts are gazed later when humans gaze at an image, depending not only on the content of the image but also on the gazer's appreciation and knowledge. Thus, we can expect that different users may recognize the relevant labels from the same image in different orders. Similar behavior patterns can also be observed when reading online articles. Ref. [18] shows that when people read online, they do not read word by word. Instead, they go back and forth through the article depending on their motivation, current needs, and personal characteristics. Based on these studies that have investigated the human behavior patterns, we design a novel methodology to annotate data, wherein we provide a subset of labels to the users and let them provide positive feedback if any relevant label is present in the subset. In this way, the annotation task can be simplified because the users are not expected to identify every label. The game ends as soon as they find one true label. Formally, we provide different subsets of labels (called groups of labels) to the users, and they will provide us with feedback on whether any relevant label is present in a given group for a specific instance. After collecting these group-supervised multi-label data, new problems include how do we learn to classify based on such data, and more importantly, what factors in the data collection process will impact the group-supervised multi-label learning.

As a pioneer work exploiting this problem, we propose a method for group-supervised multi-label learning, and analyze the factors that will impact the learning performance. We hope that this study can provide some insights with respect to the design of an improved crowdsourcing strategy for multi-label data. In particular, we first design a framework for multi-label learning that can generalize algorithms learning from both strongly supervised and weakly supervised data and is flexible with respect to the used loss functions and regularizers. Based on the general learning framework, we extend it to the grouped-supervised multi-label learning, and propose a learning algorithm compatible with neural networks. Thus, the techniques developed for modern deep learning, such as Adam [19] or Dropout [20], can be employed. Further, we discuss the qualitative effects of different factors, such as ambiguity degree and label coverage, on the learning performance. Such qualitatively studied factors are further explored via empirical studies. We verify the number of groups, relevant/irrelevant labels per group, and relevant/irrelevant labels covered and observe their effect on the generalization performance. Our empirical results may provide future insights with respect to advanced crowdsourcing query strategies.

The paper is organized as follows. In Section 2, we summarize related work and their relation with our studied problem. Section 3 provides the general framework of our studied problem and the implementation of our proposed method. In addition, we discuss the factors that could impact learning. The empirical studies are shown in Section 4, and the conclusion is presented in Section 5.

2 Related work

In this section, we briefly discuss multi-label learning (MLL), weakly supervised MLL, and crowdsourced MLL. Figure 1 presents an example of different types of supervision information for multi-labeled data.

2.1 Multi-label learning

The previous studies on multi-label learning [21] attempted to solve the problem by decomposing it into multiple binary classifications, i.e., one for each label. This type of method, termed as binary relevance has been criticized because it ignores the label correlations. Subsequently, various methods have been proposed considering the label correlations. For example, LabelPowerset [22] treated each subset of labels as one "label" and learned a multi-class classifier. Classifier Chain [23] achieved sequential learning in a label-by-label manner, and the prediction results of previous labels are used as features for latter training.

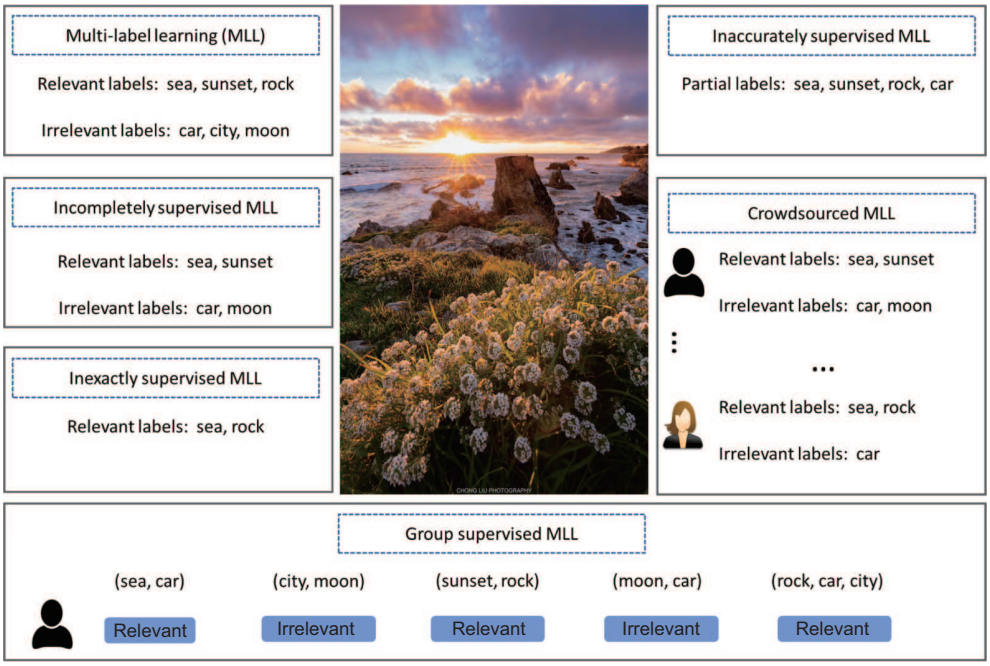

Figure 1 (Color online) Different types of supervision information for multi-labeled data.

RankSVM [24] optimized the overall learning risk for all labels plus a regularizer to incorporate the label correlation. Recently, MLL has focused on the problem of learning with extreme labels [25], wherein tens of thousands of labels are available. More information can be found in the survey study conducted by [26].

The extreme label problem can be solved via label space embedding [27–30], i.e., embedding the high-dimensional label space into a low-dimensional label space, resulting in effective testing. The embedding methods are not suitable for group-supervised MLL. They assume all the supervised information are available and learn the embedding in an ad-hoc manner; however, the given supervision is already in the grouped form in this study.

2.2 Weakly supervised MLL

To annotate a multi-class dataset, users will select only one true label. This is easy when compared with the MLL in which each label has to be checked one by one. This results in a heavy annotation load, especially when there are several labels. Thus, the weakly supervised MLL data are collected. Based on these data, the research community aims to provide robust learning algorithms. Subsequently, we briefly discuss three different types of weakly supervised MLL.

Incompletely supervised MLL. In incompletely supervised MLL, some part of the supervised information for both relevant and irrelevant labels are provided. The algorithms for incompletely supervised MLL used low rankness or manifold smoothness to regularize the label space such that the incomplete information can be compensated. For example, Refs. [9, 31–33] combined the low-rank assumption with various basic learning models for incompletely supervised MLL. Refs. [34, 35] used the smoothness assumption that similar instances should be labeled in a similar manner, such that the annotations for one instance will supplement that for another instance. In case of incompletely supervised MLL, the annotation information is provided with respect to individual labels, whereas our study does not consider an annotation for each label.

Inexactly supervised MLL. In inexactly supervised MLL, the annotation information with respect to only part of the relevant labels is given. This type of problem can be considered as a special case of incompletely supervised MLL, wherein all the irrelevant label annotations are missing. Thus, some of the methods, such as the method used by [32] to solve the incompletely supervised MLL can be generalized to solve the inexactly supervised MLL.

There are also some other methods that are specially devoted to solve the inexactly supervised MLL. For example, Ref. [11] used group lasso to find the relevant labels missed and maintain the sparseness of

the learned annotations. Ref. [10] assumed that the classification hyperplane moved across the low-density regions and proposed an algorithm based on the fact that the number of relevant labels is considerably less than that of the irrelevant labels. The annotation information on individual relevant labels is also available in this case, whereas it is not available in case of group-supervised MLL.

Inaccurately supervised MLL. Partial MLL [12] is a situation of inaccurately supervised MLL, in which a set of candidate labels is given for each instance, and the candidate labels contain at least one of the true labels. This type of learning is different from the group-supervised MLL for two aspects. One aspect is that in case of inaccurately supervised MLL, only one set of supervision information is provided per instance, whereas multiple sets of supervision information are given in this study. In addition, information about negative groups is available besides information about positive groups in group-supervised MLL.

2.3 Crowdsourced MLL

In crowdsourced MLL, the instances are distributed to multiple annotators (or users) and each annotator will annotate according to their own expertise. Ref. [15] comprehensively investigated this problem, and proposed two algorithms, among which one learns the user expertise and uses this expertise together with their annotations to learn a classifier. They further proposed a method to collect annotations actively by querying the user with labels agreeing well with their expertise.

Ref. [15] conducted a simple user behavior study based on the collected data. Further, they concluded that users would only select some relevant labels that they are familiar with. This type of study is one motivation of this study to design a new annotation strategy that could potentially reduce the labeling cost associated with future crowdsourcing tasks. Unlike [15], this study also focuses on learning directly from the group-supervised data; thus, this study does not involve the estimation of user expertise. This study may be combined with the expertise estimation in [15] to improve the crowdsourced MLL.

3 Group-supervised MLL

In this section, we present the formulation of the studied problem. Then, a general optimization objective is presented for MLL. This objective is extended to the grouped-supervised MLL problem. This objective is quite general, and various loss functions and regularizers can be used. This objective is also flexible to be optimized using various basic learning models. We implement an algorithm that optimizes this objective using neural networks because of their ability in achieving satisfactory empirical performance.

Further, we qualitatively discuss the factors that may impact the learning performance, and present a probabilistic generation model for the group-supervised MLL data based on the discussed factors.

3.1 Formulation

Let us assume we have instances space $\mathcal{X} \subset \mathbb{R}^d$ and label space $\mathcal{Y} = \{0,1\}^m$, where d is the number of features and m is the number of labels. In case of strongly supervised MLL, we have a dataset $\mathcal{D} = \{(\boldsymbol{x}_1, \boldsymbol{Y}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{Y}_n)\}$, the element $(\boldsymbol{x}_i, \boldsymbol{Y}_i)$ of which is drawn i.i.d. from $\mathcal{X} \times \mathcal{Y}$. If the jth label is relevant for the instance $\boldsymbol{x}_i$, then $Y_{ij} = 1$; otherwise, 0. In case of MLL, the target is to learn a classifier $\boldsymbol{f} : \mathcal{X} \to \mathcal{Y}$. This target is usually relaxed to learn a classifier with real-valued outputs, i.e., $\boldsymbol{g} : \mathcal{X} \to \mathbb{R}^m$, such that the jth element of $\boldsymbol{g}(\boldsymbol{x})$, i.e., $g_j(\boldsymbol{x})$ measures the relevance of the jth label with respect to the instance $\boldsymbol{x}$.

In grouped-supervised MLL, we consider a simplified case, wherein an oracle will always provide the correct annotation. Thus, we propose an algorithm to learn a classifier for MLL from such data, and discuss the potential factors that may impact the performance of the learned classifier. In particular, in case of $\boldsymbol{x}_i$, the supervision information $\boldsymbol{S}^i$ is provided, where $\boldsymbol{S}^i = \{(S_1^i, y_1^i), (S_2^i, y_2^i), \ldots, (S_{K_i}^i, y_{K_i}^i)\}$. Here, K_i is the number of groups for instance $\boldsymbol{x}_i$. $S_k^i \in [m]$ is the group of labels, i.e., a subset of all the indices of the labels. The label group S_k^i with a label $y_i^k \in \{0,1\}$ suggests the existence of any positive label within the group. Hence, y_k^i for $\boldsymbol{S}^i$ is determined by $\boldsymbol{Y}_i$, i.e.,

$$y_k^i = \begin{cases} 1, & \sum_{j \in S_k^i} Y_{ij} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

In the problem setting, the groups $\{S_1, \ldots, S_K\}$ are different for different instances. Thus, our proposal is different from MLL studies on ECOC [36].

3.2 Learning objective for MLL

Similar to the majority of learning problems, we want to learn the classifier $\boldsymbol{g} \in \mathcal{G}$ by minimizing the classification risk associated with MLL.

$$\mathcal{R}(\boldsymbol{g}) = \mathbb{E}_{(\boldsymbol{X},\boldsymbol{Y})}[\ell(\boldsymbol{g}(\boldsymbol{X}), \boldsymbol{Y})],$$

where $\ell(\cdot,\cdot)$ must be a proper loss that is continuous, non-negative, and zero when we achieve accurate predictions. Ideally, we have

$$\boldsymbol{g}^* = \arg\min_{\boldsymbol{g}\in\mathcal{G}} \mathcal{R}(\boldsymbol{g}) \tag{2}$$

as the optimal classifier that we can learn. However, we usually obtain our classifier through empirical risk minimization because we cannot have an accurate expectation in the presence of limited data, i.e.,

$$\widehat{\boldsymbol{g}}^* = \arg\min \widehat{\mathcal{R}}(\boldsymbol{g}) = \arg\min_{\boldsymbol{g}\in\mathcal{G}} \sum_{i=1}^{n} \ell(\boldsymbol{g}(\boldsymbol{x}_i), \boldsymbol{Y}_i). \tag{3}$$

Practically, we realize learning by minimizing the following objective to control the complexity of the learned model, ensuring that overfitting is prevented or the label correlation associated with MLL is considered.

$$\min \sum_i \ell(f(\boldsymbol{x}_i), \boldsymbol{Y}_i) + \lambda\Omega(\boldsymbol{f}), \tag{4}$$

where $\Omega(\cdot)$ is the regularizer and λ is the trade-off parameter.

Eq. (4) is a general objective for learning multi-labeled data, and many classical methods, regardless of strongly supervised or weakly supervised MLL, can be generalized to learning with such an objective. For example, the classical work for MLL RankSVM [24] optimizes the following objective:

$$\min \sum_i \sum_{j_1:Y_{i,j_1}=1} \sum_{j_2:Y_{i,j_2}=0} -(\boldsymbol{w}_{j_1}\boldsymbol{x} - \boldsymbol{w}_{j_2}\boldsymbol{x}) + \lambda \sum_j \|\boldsymbol{w}_j\|^2,$$

where $\boldsymbol{w}_j$ is the coefficient of the linear classifier for the jth label. Another example is incomplete MLL method Maxide [9], which learns through the following optimization objective:

$$\min \sum_{(i,j)\in\Omega_0} \left(g_j(\boldsymbol{x}_i) - Y_{ij}\right)^2 + \lambda \left\|[\boldsymbol{g}(\boldsymbol{x}_1); \ldots; \boldsymbol{g}(\boldsymbol{x}_n)]\right\|_{\mathrm{tr}}, \tag{5}$$

where Ω_0 contains the indices of the observed instance-label pairs and $\|\cdot\|_{\mathrm{tr}}$ is the spectral norm of a matrix.

Given the generalization of (4), we will attempt to extend such a generalized learning objective for MLL to group-supervised MLL.

3.3 Learning objective for group-supervised MLL

In group-supervised MLL, we have an objective similar to (4), i.e., learning through empirical risk minimization:

$$\min \sum_i \mathcal{L}(\boldsymbol{g}(\boldsymbol{x}_i), \boldsymbol{S}^i) + \lambda\Omega(\boldsymbol{g}),$$

where $\mathcal{L}(\cdot,\cdot)$ is also a proper loss function defined specially for group-supervised MLL. We assume that $\mathcal{L}(\cdot,\cdot)$ can be decomposed into each group, i.e.,

$$\mathcal{L}(\boldsymbol{g}(\boldsymbol{x}_i), \boldsymbol{S}^i) = \mathcal{L}_k(\boldsymbol{g}(\boldsymbol{x}_i), (S_k^i, y_k^i)).$$

If we assume the loss function in (4) can also be decomposed onto each label, such as that in (5), we obtain

$$\ell(\boldsymbol{g}(\boldsymbol{x}_i), \boldsymbol{Y}_i) = \ell_j(g_j(\boldsymbol{x}_i), Y_{ij}).$$

Considering the relation between the group label y_k^i and the original label $\boldsymbol{Y}_i$, we can rewrite (1) as

$$y_k^i = \min_{j \in S_k^i} Y_{ij},$$

which implies that we only need to care about the relevant labels in S_k^i if $y_k^i = 1$. However, all the labels in S_k^i are irrelevant when $y_k^i = 0$, and we can incorporate all of them into the learning process to save the provided information. Thus, we establish a connection between the MLL loss function $\ell_j(\cdot, \cdot)$ and the group-supervised MLL loss function $\mathcal{L}_k(\cdot, (\cdot, \cdot))$,

$$\mathcal{L}_k(\boldsymbol{g}(\boldsymbol{x}), (S_k, y_k)) = y_k \min_{j \in S_k} \ell_j(g_j(\boldsymbol{x}_i), +1) + (1 - y_k) \sum_{j \in S_k} \ell_j(f_j(\boldsymbol{x}_i), 0).$$

In addition to the above definition, we can assign different weights to different groups of labels according to practical requirements when either relevant or irrelevant labels are more important. In the current version, we omit the weights to clearly illustrate the effect optimizing this loss function.

With the loss function $\mathcal{L}(\cdot, \cdot)$ defined for group-supervised MLL, we can obtain the optimal classifier $\boldsymbol{g}_S$ as

$$\boldsymbol{g}_S^* = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \mathcal{R}_S(\boldsymbol{g}) = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{S})} \left[\mathcal{L}(\boldsymbol{g}(\boldsymbol{x}), \boldsymbol{S})\right] = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \mathbb{E}_{(\boldsymbol{x}, \boldsymbol{S})} \left[\sum_k \mathcal{L}_k(\boldsymbol{g}(\boldsymbol{x}), (S^k, y^k))\right], \tag{6}$$

and the same empirical risk minimization principle can be applied to obtain

$$\widehat{\boldsymbol{g}}_S^* = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \widehat{\mathcal{R}}_S(\boldsymbol{g}) = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \sum_i \left(\mathcal{L}(\boldsymbol{g}(\boldsymbol{x}_i), \boldsymbol{S}^i)\right) = \arg\min_{\boldsymbol{g} \in \mathcal{G}} \sum_i \left(\sum_k \mathcal{L}_k(\boldsymbol{g}(\boldsymbol{x}_i), (S_i^k, y_i^k))\right). \tag{7}$$

A regularizer can be also added to reduce overfitting practically and consider the correlation between labels, resulting in our learning objective of

$$\sum_i \left(\sum_k \mathcal{L}_k(\boldsymbol{g}(\boldsymbol{x}_i), (S_i^k, y_i^k))\right) + \lambda \Omega(\boldsymbol{g}). \tag{8}$$

Remark. The theoretical properties of the proposed learning objective are worth exploring. One property is how the optimal algorithm $\boldsymbol{g}_S^*$ acquired through (6) is different from that acquired through (2). In other words, when sufficient data are available, how will the classifier learned from group supervision be different from the normal MLL classifier? Another property is related to the estimation error, i.e., the difference between the empirical optimal classifier $\widehat{\boldsymbol{g}}_S^*$ and the optimal classifier $\boldsymbol{g}_S^*$. A recent study [37] on multi-class partial label learning proposes a loss function based on the min operator. Insights can be obtained based on their studies with respect to the theoretical properties of the minimizer for the loss function. We will consider such types of studies in the future.

3.4 An implementation through neural networks

Based on the learning objective, i.e., Eq. (8), we propose an algorithm named group-supervised multi-label learning GS-MLL using neural networks. The procedures are summarized in Algorithm 1.

Here we explicitly write the parameterized $\boldsymbol{g}$ as $\boldsymbol{g}(\boldsymbol{x}; \Theta)$, where Θ indicates the parameters associated with the neural networks, including the coefficients and biases. The dataset used can be applied to determine the structured of the neural networks. We select simple networks with one hidden layer for simple datasets. Deeper and wider neural networks can be used for large datasets. We will learn Θ using gradient descend optimizers. Line 1 of the algorithm is to set the optimizer $\mathcal{A}$ that can be either stochastic gradient descent (SGD) [38], Adam [19], or L-BFGS [39]. We shuffle the data into B mini batches, compute the empirical risk associated with each mini batch, calculate the gradient, and perform gradient descent accordingly using the given optimizer. Such an algorithm is flexible with respect to the loss function used. In this study, we use the simplest mean squared error as the incomplete supervised MLL [9]. For the regularizer, we use the L2 loss on the coefficients for ensuring simplicity.

Algorithm 1 GS-MLL: group-supervised multi-label learning

Input: the training set $\mathcal{D} = \{(\boldsymbol{x}_1, \boldsymbol{S}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{S}_n)\}$; epoch number T; number of mini batches B;
Output: Θ, the model parameter for $\boldsymbol{g}(\boldsymbol{x}; \Theta)$;

```
Let A be any stochastic optimization algorithm;
t = 1;
while t ⩽ T do
    t = t + 1;
    Shuffle D into B mini-batches;
    b = 1;
    while b ⩽ B do
        Pick the bth mini batch;
        Compute the empirical risk L on the mini batch by (8);
        Calculate the gradient −∇_Θ L;
        Update Θ by A;
        b = b + 1;
    end while
end while
```

3.5 Factors impacting learning

Here we discuss the factors that will impact learning. The supervision information deficiency determines the learning performance. The learning performance deteriorates with the increasing information loss. This intuition has been verified by empirical studies in other weakly supervised MLL works, such as [9]. Thus, in this subsection, we qualitatively discuss the factors that determine the supervision deficiency associated with group-supervised MLL.

3.5.1 *Ambiguity degree*

The first type of information deficiency can be observed within each group. There are two types of groups. In the positive groups, i.e., $y = 1$, the information deficiency depends on the number of relevant and irrelevant labels contained in the group. The information contained in the group increases with the increasing number of relevant labels and decreasing number of irrelevant labels. Thus, the ambiguity of a positive group can be considered as one factor impacting the performance. On the other hand, in case of negative groups, no ambiguity can be observed because all of the labels in the groups are negative. Thus, we first define the ambiguity degree for a positive group.

Definition 1 (Ambiguity degree for positive group)**.** For $(\boldsymbol{x}_i, \boldsymbol{S}^i)$ with unobserved supervised information $\boldsymbol{Y}_i$, the ambiguity degree for positive group δ_k^i for the ith instance of a positive group $(S_k^i, 1) \in \boldsymbol{S}^i$ is defined as

$$\delta_k^i = \frac{\sum_{j \in S_k} Y_j}{|S_k|}.$$

The ambiguity degree for a particular instance is defined as follows.

Definition 2 (Ambiguity degree for an instance)**.** For $(\boldsymbol{x}_i, \boldsymbol{S}^i)$ with unobserved supervised information $\boldsymbol{Y}_i$, the ambiguity degree for an instance δ^i for the ith instance is defined as

$$\delta^i = \frac{1}{\sum_k y_k^i} \sum_k \delta_k^i,$$

where δ_k^i is the ratio of the percentage of relevant labels in a positive group.

The ambiguity degree associated with the positive group δ_k^i is dependent on the number of relevant and irrelevant labels in this positive group. In case of the ambiguity degree δ^i, it is not only dependent on each δ_k^i, but also on the number of positive groups. We will vary the number of relevant labels per positive group, the number of irrelevant labels per positive group, and the number of positive groups in the empirical studies, and observe the manner in which the performance of the group-supervised MLL is affected.

Finally, only the δ_k^i, i.e., the ratio of the percentage of relevant labels in a positive group, may not be sufficient. We also need to define the co-occurrence of relevant labels in a positive group. However, there may be no method using which two labels can be differentiated if two labels always occur together. This has been verified theoretically via studies on single-labelled partial label learning problem [40].

Definition 3 (Ambiguity degree for co-occurrence)**.** For $(\boldsymbol{x}_i, \boldsymbol{S}^i)$ with unobserved supervised information $\boldsymbol{Y}_i$, the ambiguity degree for co-occurrence ζ_k^i for the ith instance of a positive group $(S_k^i, 1) \in \boldsymbol{S}^i$ is defined as

$$\zeta_k^i = \frac{1}{\sum_j Y_{ij}} \left(\sum_{j \in S_k} Y_{ij} - 1 \right).$$

This definition is applied to each group. The label co-occurrence for instance i which is associated with multiple groups can then be defined as $\sum_k \zeta_k^i$. The definition presented in this study gives the average percentage with respect to the number of additional relevant labels in one positive group.

3.5.2 *Label coverage*

Another factor that may impact the learning of group-supervised MLL is the number of labels that are covered in the supervision information. Only those labels carried by positive groups can carry information about relevant labels. Similarly, only the labels carried by negative groups can carry information about irrelevant labels. We define two values representing the label coverages for group-supervised MLL in this study.

Definition 4 (Relevant label coverage)**.** For $(\boldsymbol{x}_i, \boldsymbol{S}^i)$ with unobserved supervised information $\boldsymbol{Y}_i$, the relevant label coverage γ_p is defined as

$$\gamma_p = \frac{1}{\sum_j Y_{ij}} \left| j : j \in \bigcup_{k: y_k^i = 1} S_k^i \text{ and } Y_{ij} = 1 \right|,$$

which is the percentage of relevant labels appearing in at least one group.

We can also define another value.

Definition 5 (Irrelevant label coverage)**.** For $(\boldsymbol{x}_i, \boldsymbol{S}^i)$ with unobserved supervised information $\boldsymbol{Y}_i$, the irrelevant label coverage γ_n is defined as

$$\gamma_n = \frac{1}{\sum_j (1 - Y_{ij})} \left| j : j \in \bigcup_{k: y_k^i = 0} S_k^i \right|$$

which is the percentage of irrelevant labels appearing in at least one negative group.

Less information about relevant labels is carried under supervision $\boldsymbol{S}$ with a small γ_p; thus, the data deficiency is critical, and it may be difficult to learn a satisfactory classifier. Similarly, the information deficiency on irrelevant labels is critical with a small γ_n, adding difficulty to learning.

Remark. In crowdsourcing tasks, the situation may be a bit different because an objective of crowdsourcing is to save labor, i.e., reduce the difficulty of the annotations task; otherwise, the annotation may be quite expensive. There exist some contradictions between annotations and learning with respect to difficulty. For example, annotators may want the co-occurrence to be high such that observing only one relevant label is sufficient to skip all the remaining relevant labels in the group. However, a high co-occurrence may deteriorate the learning performance. On the other hand, a high irrelevant label coverage is welcomed by the succeeding learning algorithms, whereas the user may need to check the labels one by one if they are all negative in a group. Therefore, there is a trade-off between the crowdsourcing and learning tasks, and such a trade-off needs to be carefully considered in the future.

3.6 Data generation model

All the aforementioned factors determine the supervision information for group-supervised MLL. In this subsection, we present a data generation process and show that the generation process considerably intervenes with the factors impacting learning.

During the sampling process, we have an instance $(\boldsymbol{x}, \boldsymbol{Y})$ based on the probability distribution $p(\boldsymbol{x}, \boldsymbol{Y})$. By considering as a condition, we need to obtain $p(\boldsymbol{x}, S_k, y_k)$. In particular, we have

$$p(\boldsymbol{x}, S_k, y_k) = \sum_{\boldsymbol{Y}} p(\boldsymbol{x}, S_k, y_k, \boldsymbol{Y}) = \sum_{\boldsymbol{Y}} p(S_k, y_k | \boldsymbol{x}, \boldsymbol{Y}) p(\boldsymbol{x}, \boldsymbol{Y}) = \sum_{\boldsymbol{Y}} p(S_k | y_k, \boldsymbol{x}, \boldsymbol{Y}) p(y_k) p(\boldsymbol{x}, \boldsymbol{Y})$$

$$= \sum_{\boldsymbol{Y}} p(S_k|y_k = 0, \boldsymbol{x}, \boldsymbol{Y})p(y_k = 0)p(\boldsymbol{x}, \boldsymbol{Y}) + p(S_k|y_k = 1, \boldsymbol{x}, \boldsymbol{Y})p(y_k = 1)p(\boldsymbol{x}, \boldsymbol{Y}), \quad (9)$$

where $p(y_k = 0)$ and $p(y_k = 1)$ are related to the number of positive and negative groups. $p(\boldsymbol{x}, \boldsymbol{Y})$ is shared with normal MLL. In case of $p(S_k|y_k = 0, \boldsymbol{x}, \boldsymbol{Y})$, we obtain

$$p(S_k|y_k = 0, \boldsymbol{x}, \boldsymbol{Y}) = \sum_{j \in S_k} p(j|y_k = 0, \boldsymbol{x}, \boldsymbol{Y}) = \sum_{j \in S_k} p(j|y_k = 0, \boldsymbol{x}, Y_{ij} = 0),$$

where $p(j|y_k = 0, \boldsymbol{x}, Y_{ij} = 0)$ corresponds to the irrelevant label coverage γ_n.

Another probability $p(S_k|y_k = 1, \boldsymbol{x}, \boldsymbol{Y})$ can be written as

$$p(S_k|y_k = 1, \boldsymbol{x}, \boldsymbol{Y}) = \sum_{j \in S_k} p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 0) + p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 1),$$

where $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 1)$ corresponds to the relevant label coverage γ_p. The ratio between $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 0)$ and $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 1)$ gives the ambiguity degree for positive group δ_k^i. Finally, the ambiguity degree for co-occurrence ζ_k^i is decided by both $\sum_j Y_{ij}$ and $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 1)$.

The process to generate the group-supervised MLL data is summarized as follows. We first generate $(\boldsymbol{x}, \boldsymbol{Y})$ according to $p(\boldsymbol{x}, \boldsymbol{Y})$. Then we decide whether to generate a group with label y_k through $p(y_k)$. After determining the label of a group, we generate either a positive group or a negative group. If the group is negative, we generate the group label by label according to $p(j|y_k = 0, \boldsymbol{x}, Y_{ij} = 0)$. Otherwise, we generate relevant and irrelevant labels through $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 1)$ and $p(j|y_k = 1, \boldsymbol{x}, Y_{ij} = 0)$, respectively, in a label-by-label manner. This generation process will be presented in Section 4 for empirical studies.

4 Experiments

In this section, we will show the performance of our proposed method using real-world dataset and the manner in which different factors affect the learning performance.

Data generation. We generate the group-supervised MLL datasets through a benchmark data set Yeast [24], which contains 2417 instances and 14 labels according to the data generation model in Subsection 3.6. On an average, each instance in Yeast contains more than 4 relevant labels different from other multi-label datasets that only have a small number of relevant labels, making it a suitable medium-sized dataset for analyzing the factors impacting learning.

We first set the number of positive and negative groups to generate the data corresponding to $p(y_k)$ in (9). Without losing generality, we set the number of negative groups to be 1 because all the labels inside will be considered as irrelevant labels. We vary the number of positive groups to be within $\{2, 3, 4\}$ and verify their impacts on performance.

One important factor required to generate a negative group is the irrelevant label coverage. Here, we sample a part of all the irrelevant labels to be included in the negative group, and vary the ratio to be within $\{0.25, 0.5, 0.75, 1.0\}$ and observe their impacts on the learning performance.

To generate the positive group, we first consider the relevant labels in the positive group. We control the relevant label coverage to be within $\{0.25, 0.5, 0.75, 1.0\}$ based on which a candidate set for relevant labels is generated. By sampling uniformly at random from the candidate set, the positive group has its relevant labels. We control the number of relevant labels per positive group to be within $\{1, 2\}$. Finally, we sample uniformly at random from all the irrelevant labels that have to be added to the positive group. The number of irrelevant labels is varied within $\{1, 2, 3\}$.

For each generation parameter, we generate 10 data using different random seeds, such that the obtained results are the average of 10 results. The dataset is split into training and testing parts, with 1500 training instances and 934 test instances. After training the classifier using the training data, the performances on the test instances are reported.

Settings. We implement our proposed algorithm using PyTorch. In particular, we set the structure of the neural network as one hidden layer with 52 hidden neurons according to the recommendation of [41]. We add the L2 norm of coefficients as a regularizer and use the mean squared error (MSE) as our loss function. We use SGD with momentum as the optimizer, and the momentum parameter is set as 0.9. Subsequently, we perform cross-validation to select the best step size and weight decay parameter

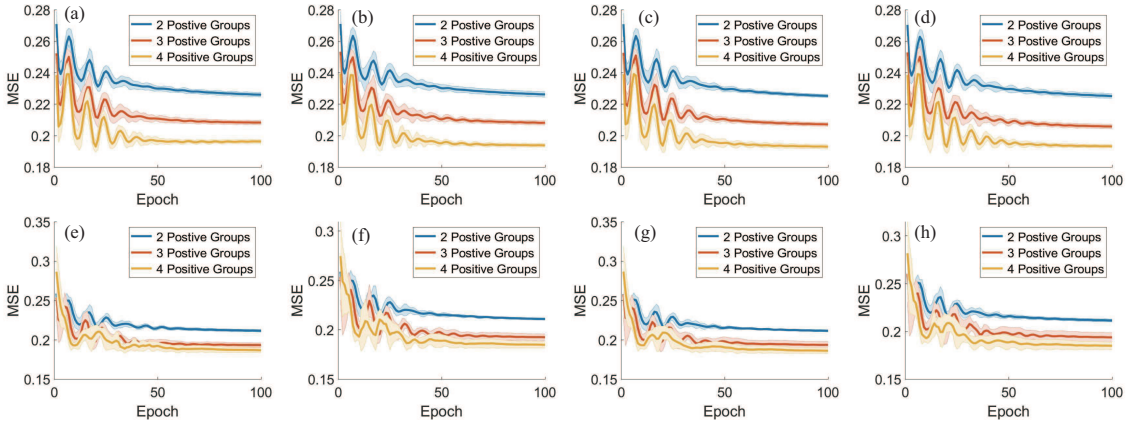

Figure 2 (Color online) Test results show how the number of positive groups impacts learning. x-axis is the epoch, and y-axis is the MSE. The shadow area shows the standard deviation (STD) of ten random trials under the same setting. ir is the number of irrelevant labels in each positive group, and rc shows the relevant label coverage. The number of relevant label per group is fixed to be 1, and the irrelevant label coverage is fixed to be 0.5. (a) ir: 2, rc: 0.25; (b) ir: 2, rc: 0.5; (c) ir: 2, rc: 0.75; (d) ir: 2, rc: 1.0; (e) ir: 4, rc: 0.25; (f) ir: 4, rc: 0.5; (g) ir: 4, rc: 0.75; (h) ir: 4, rc: 1.0.

from $\{10^{-10}, 10^{-9}, \ldots, 10^{5}\}$. Finally, we present the performance obtained with respect to the MSE. We also evaluated our proposed algorithm using classical multi-label measurements such as hamming loss or ranking loss. Since they show the same trend as MSE, we report only MSE in this study because of space limitation.

4.1 Number of positive groups

In this subsection, we fix the remaining factors and observe the manner in which the variation of the number of positive groups affects the learning performance. The results are shown in Figure 2. We obtain 96 different results. Here, we only show 8 of them because of space limitation. We fix the number of relevant labels per group to be 1 and the irrelevant label coverage to be 1/2. Subsequently, we varied the remaining values. Based on the results, the performance is observed to improve with the increasing number of positive groups when the remaining conditions are fixed. However, when the number of irrelevant labels reaches 4, the advantage obtained by increasing the number of positive groups becomes marginal.

4.2 Relevant label coverage

In this subsection, we fix the remaining factors and observe the manner in which the variation of the relevant label coverage can affect the learning performance. We report 8 experimental results, wherein the number of positive groups is fixed to be 3 and the number of relevant labels per group is fixed to be 1. Then, we vary the number of irrelevant labels per group and the irrelevant label coverage. Figure 3 shows the results. The results indicate that contradictory to our intuition, the relevant label coverage does not considerably affect the learning performance.

4.3 Irrelevant label coverage

In this subsection, we fix the remaining factors and observe the manner in which the variation of the irrelevant label coverage can affect the learning performance. We report 8 experimental results by fixing the number of irrelevant labels per positive group to be 2 and the number of relevant labels per group to be 1. Then, we vary the number of positive groups and the relevant label coverage. The results are shown in Figure 4. Thus, the results are observed to be contradictory to our intuition, i.e., the performance deteriorates with the increasing number of irrelevant labels in the negative group. We intend to further explore this interesting observation in the future.

4.4 Relevant labels per positive group

We analyze the manner in which the number of relevant labels per positive group affects learning. Here, we fix the number of positive groups to be 3 and the relevant label coverage to be 0.5. Further, we

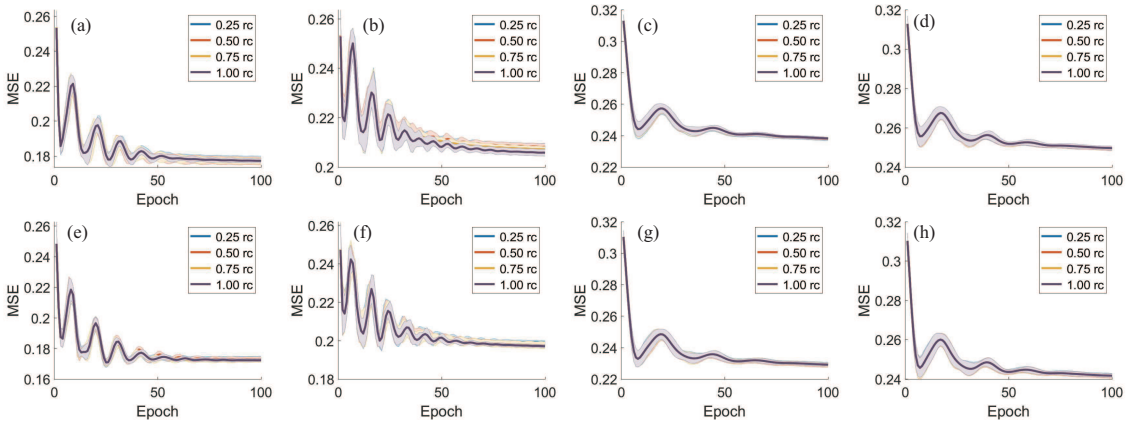

Figure 3 (Color online) Test results show how the relevant label coverage (rc) impacts learning. x-axis is the epoch, and y-axis is the MSE. Under the same setting, the shadow area shows the variance of ten random trials. ir is the number of irrelevant labels in each positive group, and ic shows the irrelevant label coverage. The number of relevant label per group is fixed to be 1, and the number of positive groups is fixed to be 3. (a) ir: 1, ic: 0.25; (b) ir: 1, ic: 0.5; (c) ir: 1, ic: 0.75; (d) ir: 1, ic: 1.0; (e) ir: 2, ic: 0.25; (f) ir: 2, ic: 0.5; (g) ir: 2, ic: 0.75; (h) ir: 2, ic: 1.0.

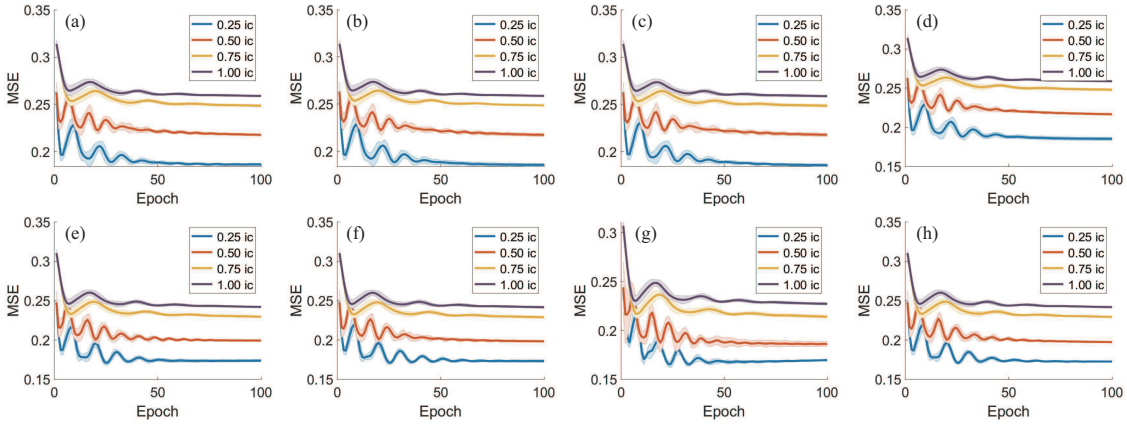

Figure 4 (Color online) Test results show how the irrelevant label coverage (ic) impacts learning. x-axis is the epoch, and y-axis is the MSE. Under the same setting, the shadow area shows the STD of ten random trials. ng is the number of positive groups, and rc shows the relevant label coverage. The number of relevant label per group is fixed to be 1, and the number of irrelevant label per group is fixed to be 2. (a) ng: 2, rc: 0.25; (b) ng: 2, rc: 0.5; (c) ng: 2, rc: 0.75; (d) ng: 2, rc: 1.0; (e) ng: 3, rc: 0.25; (f) ng: 3, rc: 0.5; (g) ng: 3, rc: 0.75; (h) ng: 3, rc: 1.0.

vary the irrelevant label coverage and the number of irrelevant labels per positive group. Figure 5 shows the results. Here, the result is contradictory to our intuition, i.e., the performance improves with the increasing number of relevant labels per group.

4.5 Irrelevant labels per positive group

In this subsection, we present the manner in which the number of irrelevant labels per positive group impacts learning. Here, we fix the number of relevant labels per positive group to be 1 and the relevant label coverage to be 0.5. Then we vary the irrelevant label coverage and the number of positive groups. Figure 6 shows the results. Here, we were surprised to observe that the performance improved with the increasing number of irrelevant labels in the relevant label set.

Summary. Based on the above experimental results, the performance will improve with the increasing number of labels in the positive label group, regardless of whether they are relevant or irrelevant. However, covering more irrelevant labels in a negative group may not result in better performance. Moreover, the relevant label coverage only exhibits a minor impact.

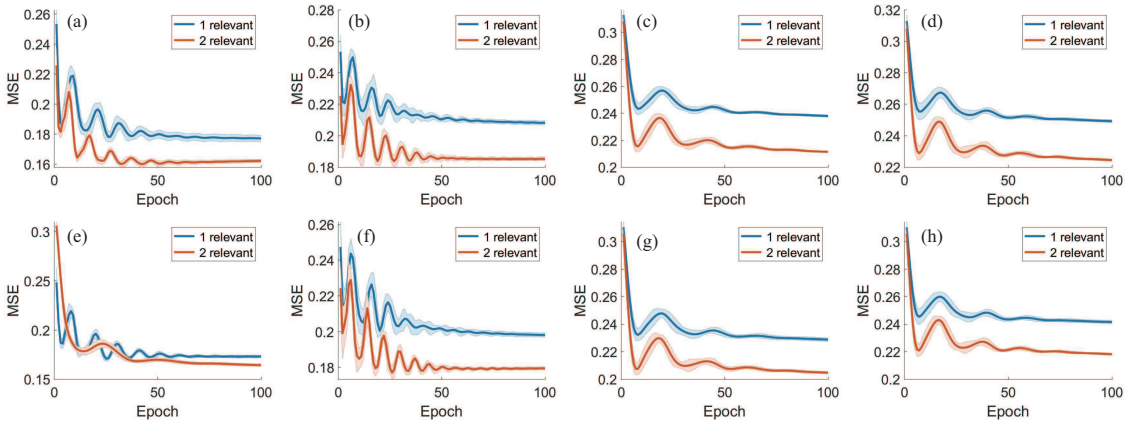

Figure 5 (Color online) Test results show how the number of relevant labels per positive group impacts learning. x-axis is the epoch, and y-axis is the MSE. Under the same settings, the shadow area shows the STD of ten random trials. ir is the number of irrelevant labels per positive group, and ic shows the irrelevant label coverage. The number of positive groups and the relevant label coverage are fixed to be 3 and 0.5 respectively. (a) ir: 1, ic: 0.25; (b) ir: 1, ic: 0.5; (c) ir: 1, ic: 0.75; (d) ir: 1, ic: 1.0; (e) ir: 2, ic: 0.25; (f) ir: 2, ic: 0.5; (g) ir: 2, ic: 0.75; (h) ir: 2, ic: 1.0.

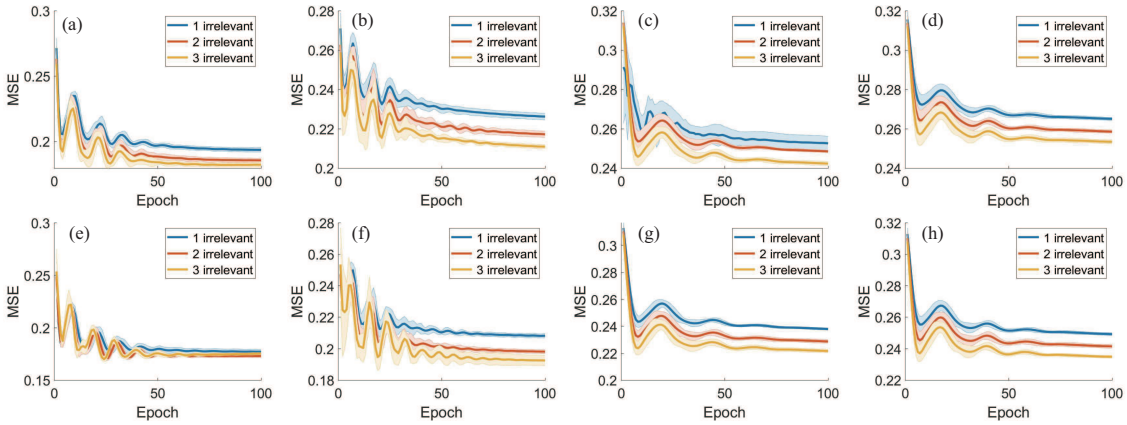

Figure 6 (Color online) Test results show how the number of irrelevant labels per positive group impacts learning. x-axis is the epoch, and y-axis is the MSE. Under the same setting, the shadow area shows the STD of ten random trials. ng is the number of positive groups, and ic shows the irrelevant label coverage. The number of relevant labels per group and the relevant label coverage are fixed to be 1 and 0.5, respectively. (a) ng: 2, ic: 0.25; (b) ng: 2, ic: 0.5; (c) ng: 2, ic: 0.75; (d) ng: 2, ic: 1.0; (e) ng: 3, ic: 0.25; (f) ng: 3, ic: 0.5; (g) ng: 3, ic: 0.75; (h) ng: 3, ic: 1.0.

5 Conclusion

In this study, we investigate the problem of group-supervised MLL, wherein labels are organized into groups. One group is considered to be positive if it contains any relevant label, and negative if it does not contain any relevant label. This problem setting is motivated by crowdsourced MLL and human behaviour pattern studies conducted by neuroscientists. Based on the collected data, we propose an algorithm based on neural networks and discuss the potential factors that may impact learning. We finally show empirically how the learning performance is impacted. Our empirical observations provide some interesting results that are contradictory to our assumptions. These results may provide insight for realizing future crowdsourced MLL. We also plan to explore the manner in which the proposed method will perform when using other loss functions, inlcuding logistic loss and sigmoid loss.

References

1 Zhou Z-H, Zhang M-L. Multi-label learning. In: Encyclopedia of Machine Learning and Data Mining. Berlin: Springer, 2016. 875–881
2 Cabral R S, de la Torre F, Costeira J P, et al. Matrix completion for weakly-supervised multi-label image classification. IEEE Trans Pattern Anal Mach Intell, 2015, 37: 121–135
3 Chen M, Zheng A X, Weinberger K Q. Fast image tagging. In: Proceedings of the 30th International Conference on Machine Learning, 2013. 1274–1282
4 Chalkidis I, Fergadiotis M, Malakasiotis P, et al. Large-scale multi-label text classification on EU legislation. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019. 6314–6322

5 Nam J, Kim J, Menciaa E L, et al. Large-scale multi-label text classification — revisiting neural networks. In: Proceedings of the 25th European Conference on Machine Learning, 2014. 437–452
6 Zhou J, Chen L, Guo Z. iATC-NRAKEL: an efficient multi-label classifier for recognizing anatomical therapeutic chemical classes of drugs. Bioinform, 2020, 36: 1391–1396
7 Zhang J, Zhang Z, Wang Z, et al. Ontological function annotation of long non-coding RNAs through hierarchical multi-label classification. Bioinform, 2018, 34: 1750–1757
8 Zhou Z H. A brief introduction to weakly supervised learning. Natl Sci Rev, 2018, 5: 44–53
9 Xu M, Jin R, Zhou Z. Speedup matrix completion with side information: application to multi-label learning. In: Proceedings of Advances in Neural Information Processing Systems 26, 2013. 2301–2309
10 Sun Y, Zhang Y, Zhou Z. Multi-label learning with weak label. In: Proceedings of the 24th Conference on Artificial Intelligence, 2010
11 Bucak S S, Jin R, Jain A K. Multi-label learning with incomplete class assignments. In: Proceedings of the 24th Conference on Computer Vision and Pattern Recognition, 2011. 2801–2808
12 Xie M, Huang S. Partial multi-label learning. In: Proceedings of the 32nd Conference on Artificial Intelligence, 2018. 4302–4309
13 Yu G, Chen X, Domeniconi C, et al. Feature-induced partial multi-label learning. In: Proceedings of the 2018 International Conference on Data Mining, 2018. 1398–1403
14 Estelles-Arolas E, González-Ladrón-de-Guevara F. Towards an integrated crowdsourcing definition. J Inf Sci, 2012, 38: 189–200
15 Li S, Jiang Y, Chawla N V, et al. Multi-label learning from crowds. IEEE Trans Knowl Data Eng, 2019, 31: 1369–1382
16 Li S, Jiang Y. Multi-label crowdsourcing learning with incomplete annotations. In: Proceedings of the 15th Pacific Rim International Conference on Artificial Intelligence, 2018. 232–245
17 Quiroga R Q, Pedreira C. How do we see art: an eye-tracker study. Front Hum Neurosci, 2011, 5: 98
18 Group N N. How People Read Online: The Eyetracking Evidence. 2nd ed. Technical Report, 2020
19 Kingma D P, Ba J. Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations, 2015
20 Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res, 2014, 15: 1929–1958
21 Boutell M R, Luo J, Shen X, et al. Learning multi-label scene classification. Pattern Recogn, 2004, 37: 1757–1771
22 Tsoumakas G, Vlahavas I P. Random k-labelsets: an ensemble method for multilabel classification. In: Proceedings of the 18th European Conference on Machine Learning, 2007. 406–417
23 Read J, Pfahringer B, Holmes G, et al. Classifier chains for multi-label classification. In: Proceedings of the 20th European Conference on Machine Learning, 2009. 254–269
24 Elisseeff A, Weston J. A kernel method for multi-labelled classification. In: Proceedings of Advances in Neural Information Processing Systems 14, 2001. 681–687
25 Bhatia K, Jain H, Kar P, et al. Sparse local embeddings for extreme multi-label classification. In: Proceedings of Advances in Neural Information Processing Systems 28, 2015. 730–738
26 Zhang M L, Zhou Z H. A review on multi-label learning algorithms. IEEE Trans Knowl Data Eng, 2014, 26: 1819–1837
27 Hsu D J, Kakade S M, Langford J, et al. Multi-label prediction via compressed sensing. In: Proceedings of Advances in Neural Information Processing Systems 22, 2009. 772–780
28 Tai F, Lin H T. Multilabel classification with principal label space transformation. Neural Computation, 2012, 24: 2508–2542
29 Bi W, Kwok J T. Efficient multi-label classification with many labels. In: Proceedings of the 30th International Conference on Machine Learning, 2013. 405–413
30 Ubaru S, Mazumdar A. Multilabel classification with group testing and codes. In: Proceedings of the 34th International Conference on Machine Learning, 2017. 3492–3501
31 Goldberg A B, Zhu X, Recht B, et al. Transduction with matrix completion: three birds with one stone. In: Proceedings of Advances in Neural Information Processing Systems 23, 2010. 757–765
32 Bi W, Kwok J T. Multilabel classification with label correlations and missing labels. In: Proceedings of the 28th Conference on Artificial Intelligence, 2014. 1680–1686
33 Xu L, Wang Z, Shen Z, et al. Learning low-rank label correlations for multi-label classification with missing labels. In: Proceedings of the 2014 International Conference on Data Mining, 2014. 1067–1072
34 Jing L, Yang L, Yu J, et al. Semi-supervised low-rank mapping learning for multi-label classification. In: Proceedings of the 28th Conference on Computer Vision and Pattern Recognition, 2015. 1483–1491
35 Wu B, Lyu S, Ghanem B. ML-MG: multi-label learning with missing labels using a mixed graph. In: Proceedings of the 2015 International Conference on Computer Vision, 2015. 4157–4165
36 Ferng C, Lin H. Multi-label classification with error-correcting codes. In: Proceedings of the 3rd Asian Conference on Machine Learning, 2011. 281–295
37 Lv J, Xu M, Feng L, et al. Progressive identification of true labels for partial-label learning. 2020. ArXiv:2002.08053
38 Bottou L. On-line learning and stochastic approximations. In: Online Learning in Neural Networks. Cambridge: Cambridge University Press, 1998. 9–42
39 Andrew G, Gao J. Scalable training of L1-regularized log-linear models. In: Proceedings of the 24th International Conference on Machine Learning, 2007. 33–40
40 Liu L, Dietterich T G. Learnability of the superset label learning problem. In: Proceedings of the 31st International Conference on Machine Learning, 2014. 1629–1637
41 Zhang M-L, Zhou Z-H. Multilabel neural networks with applications to functional genomics and text categorization. IEEE Trans Knowl Data Eng, 2006, 18: 1338–1351