SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2025, Vol. 68, Iss. 6, 162102:1–162102:18
https://doi.org/10.1007/s11432-023-3979-9

Pushing one pair of labels apart each time in multi-label learning: from single positive to full labels

Xiang LI†, Xinrui WANG† & Songcan CHEN*

MIIT Key Laboratory of Pattern Analysis and Machine Intelligence, College of Computer Science and Technology/ College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Received 26 July 2023/Revised 7 October 2023/Accepted 30 January 2024/Published online 20 January 2025

Abstract In multi-label learning (MLL), it is extremely challenging to accurately annotate every appearing object due to expensive costs and limited knowledge. When facing such a challenge, a more practical and cheaper alternative should be single positive multi-label learning (SPMLL), where only one positive label needs to be provided per sample. Existing SPMLL methods usually assume unknown labels as negatives, which inevitably introduces false negatives as noisy labels. More seriously, binary cross entropy (BCE) loss is often used for training, which is notoriously not robust to noisy labels. To mitigate this issue, we customize an objective function for SPMLL by pushing only one pair of labels apart each time to suppress the domination of negative labels, which is the main culprit of fitting noisy labels in SPMLL. To further combat such noisy labels, we explore the high-rankness of the label matrix, which can also push apart different labels. By directly extending from SPMLL to MLL with full labels, a unified loss applicable to both settings is derived. As a byproduct, the proposed loss can alleviate the imbalance inherent in MLL. Experiments on real datasets demonstrate that the proposed loss not only performs more robustly to noisy labels for SPMLL but also works well for full labels. Besides, we empirically discover that high-rankness can mitigate the dramatic performance drop in SPMLL. Most surprisingly, even without any regularization or fine-tuned label correction, only adopting our loss defeats state-of-the-art SPMLL methods on CUB, a dataset that severely lacks labels.

Keywords multi-label learning, single positive label, noisy labels, missing labels, image classification

Citation Li X, Wang X R, Chen S C. Pushing one pair of labels apart each time in multi-label learning: from single positive to full labels. Sci China Inf Sci, 2025, 68(6): 162102, https://doi.org/10.1007/s11432-023-3979-9

1 Introduction

As a general extension of multi-class learning, multi-label learning (MLL) [1–3] often contains multiple labels in a single training sample, and the goal is to assign every label associated with the corresponding sample. Since the setting of MLL is much closer to reality where multi-objects often co-occur in a natural scene, it has received considerable attention in the past two decades and has developed wide applications as diverse as image classification [4–8], video analysis [9–12], natural language processing [13–16], just to name a few.

As we all know, exhaustively and accurately annotating every object that appears in a sample, i.e., making full labels, is extremely expensive, and sometimes even impossible, since there may exist unknown objects due to limited knowledge. To circumvent the obstacles in making full labels, a good and more practical alternative should be single positive multi-label learning (SPMLL) [17]; i.e., annotating only one object that appears in a sample and other labels remain unknown or unannotated.

The advantages of SPMLL lie in the following aspects. First, annotating single positive label is much simpler and less expensive since there is no need to exhaustively examine every corner of the sample. Second, the risk of introducing false positive label is greatly reduced because we tend to annotate the object that we are more familiar with and more confident in. Recently, Ref. [18] has proven that the single positive label is sufficient for MLL, which provides theoretical support for this application.

Although SPMLL enjoys the above advantages, naturally, the least annotation of each sample tends to make traditional MLL methods underperform. And those relying on the co-occurrence to learn label correlations [19–22] will even fail due to that the co-occurrence in a single sample is no longer available

* Corresponding author (email: s.chen@nuaa.edu.cn)
† These authors contributed equally to this work.

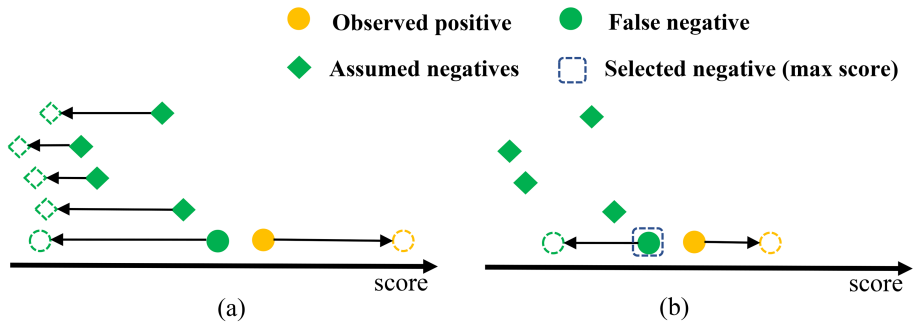

Figure 1 Optimization for (a) BCE and (b) our OPML loss. BCE loss optimizes all pairs of labels at once, whereas our loss pushes only one pair of labels apart each time. Thus, BCE can be easily dominated by negative labels while our loss can alleviate this issue.

in the setting of SPMLL. Therefore, in such an emerging field, algorithms specialized for SPMLL should be carefully designed.

To deal with the challenge of SPMLL, the pioneer work [17] assumes all the unknown labels to be negatives, and the subsequent studies [18, 23, 24] also adopted the so-called "Assume Negative" (AN) assumption. Obviously, such an assumption inevitably introduces false negatives, which involve noisy labels during training. More seriously, as the default choice of loss function in these studies, binary cross entropy (BCE) loss is notoriously not robust to noisy labels [25–27] for the reason that it treats each label equally [28, 29]. As a result, BCE loss will fit not only the clean labels but also the noisy labels. Note that in SPMLL, only one positive label is observed for each sample, and the AN assumption makes the number of negative labels far more than that of positive labels. Consequently, the negative labels will dominate the training; meanwhile, the single positive label will be underfitted during training, which naturally causes a performance gap between the SPMLL and MLL with full labels.

From the above analyses, it can be seen that there are two important questions for robust SPMLL to noisy labels, one is how to make full use of the precious single positive (clean) label, and the other is how to suppress the domination of negative labels. To address them, different from BCE loss that optimizes all pairs of labels at once, we select only one pair of labels to optimize each time, resulting in a more balanced gradient update. The key to this so-called "one pair each time" strategy is determining which labels to select for optimization. Obviously, one of the pairs should be the observed single positive label (only clean label), and the other is the unobserved label with the maximum score. Consequently, the selected pair of labels are the closest; thus pushing them apart can enhance the discriminative ability. Maximizing the difference of the scores between the selected pairs of labels encourages a large gap between them, which leads to clear separation. Such a strategy of maximizing the closest scores of labels aligns with the celebrated large margin theory [30], which explains why the discriminative ability can be enhanced. A visualized illustration of the optimization for both BCE and our loss is depicted in Figure 1 to show the difference. The main difference lies in the fact that the BCE loss pushes apart the positive label and all the "assume negative" labels simultaneously, whereas our loss pushes apart only one pair of labels each time, and all labels are optimized one by one iteratively. By such a design, the domination of negative labels is suppressed and meanwhile the importance of the positive label is consolidated. The gradient analysis in Subsection 3.3 will further verify the effectiveness of our idea.

Note that, the strategy of pushing one pair of labels apart in SPMLL can be directly extended to MLL with full labels by selecting the negative label with the maximum score and the positive label with the minimum score as a pair of labels to be optimized. With such an extension, we derive a unified loss for both SPMLL and MLL with full labels. To the best of our knowledge, such a strategy for addressing the imbalance issue has never been presented in previous studies.

As the supervised information is seriously insufficient in SPMLL, the unknown true positive label will gradually turn to a false negative in training. To further combat the noisy labels, we cooperate the high-rank property of multi-label matrix [31] with our newly proposed loss. The motivation is that the high-rank property encourages to push apart different labels, which will further slow down the process of turning the unknown true positive to false negative as shown in Subsection 4.3. Besides, experiments in Subsection 4.3 also verify the effectiveness of high-rank regularization.

In summary, our contributions are fourfold:

(1) We derive a new loss from the analysis of pushing one pair of labels apart each time for multi-label

learning abbreviated as OPML, which can be seamlessly used in both the single positive and full labels settings. As a byproduct, OPML can alleviate the imbalance inherent in MLL. Experiments show that the OPML loss not only performs more robustly to noisy labels in SPMLL but also works well in MLL with full labels.

(2) We find that the OPML loss can help discover missing ground truth labels that are neglected but actually present in the image. For more details, please refer to Subsection 4.6.

(3) We empirically discover that in SPMLL, the high-rank property can alleviate the negative impact of noisy labels during the learning process; specifically, with such regularization, the performance drop is not that dramatic, which may shed new light on general noisy label learning.

(4) We empirically verify that compared with state-of-the-art methods [23,29] that use extra regularization or fine-tuned label correction, only adopting the OPML loss can defeat them on CUB, a challenging dataset that severely lacks labels, even without any of these techniques. These results demonstrate the superiority of our OPML loss in dealing with more challenging datasets where labels are heavily scarce.

2 Related work

In this section, we review the related work from the following two aspects. The first is the commonly used loss functions in MLL and the mechanisms behind them; the second is recent advances focusing on SPMLL.

2.1 Loss functions in MLL

In both MLL and its closely related studies, BCE loss is often used as a default choice combined with various tricks and regularizations [32–35]. However, it has been shown that BCE loss is not robust to noisy labels [25–27], which are pervasive in MLL. Recently, Ref. [36] has revealed that by alleviating the imbalance problem in MLL, some variants of BCE loss like focal loss [37] and asymmetric loss [36] can significantly boost the performance. Ref. [37] was originally proposed to deal with the serious imbalance between the targets and numerous backgrounds in object detection. Since focal loss can well handle the imbalance problem, which is inherent in MLL, it has become a widely used loss in MLL [36,38–41] and has achieved great empirical success. Ref. [36] even pointed out that, by simply reducing the contribution of negative samples to the loss when their probability is low, the BCE loss with the carefully tuned reweighting parameter can reach state-of-the-art results. Besides, Ref. [42] has extended the popular cross-entropy loss to MLL by exploring the proper surrogate function for softmax in MLL, which is a special case fallen into our loss. In contrast to existing focal loss [37] and asymmetric loss [36] that assign different weights to the positive and negative samples to address the imbalance issue, our OPML loss starts from a completely new perspective, namely, only one pair of positive and negative samples is optimized each time, resulting in a more balanced gradient update. The gradient analysis in Subsection 3.3 also confirms its effectiveness. It is worth mentioning that the proposed OPML loss can be seamlessly applied in both traditional methods and popular deep neural networks.

2.2 Recent advances in SPMLL

SPMLL is first proposed by [17] considering its significantly reduced annotation costs. Due to the lack of precise supervision, it takes the AN assumption on the unobserved labels combined with various reweighting strategies by incorporating the expected number of true positive labels, which is not available in reality. Following [17], Ref. [23] also took the AN assumption and cast the SPMLL task into noisy multi-label classification. Later, instead of making the AN assumption, Ref. [29] treated all unannotated labels as unknown by maximizing the entropy, and then adopted a heuristic asymmetric pseudo-labeling method. Notably, Ref. [18] has proven that a single positive label is sufficient for MLL by deriving a risk estimator approximately converging to the optimal risk minimizer of fully supervised learning, which provides a solid theoretical support. Inspired by the empirical success of consistency regularization in multi-class classification, Ref. [24] extended this popular regularization to SPMLL with the help of their proposed label-aware attention module. Recently, Ref. [43] provided a theoretical guarantee for learning from pseudo-label on SPMLL. Plus, Ref. [44] studied the setting of SPMLL from the perspective of generating multi-label data from a single positive label. Despite great efforts that have been made, the performance gap between the SPMLL and MLL with full labels still exists.

3 Proposed approach

3.1 Problem statement

In this subsection, we first describe the setting of MLL with full labels, and then detail the setting of SPMLL.

MLL with full labels. Given a training dataset $\mathcal{D} = \{X_i, Y_i\}_{i=1}^N$, where N is the number of training samples. $Y_i \in \{0,1\}^L$, $Y_{il} = 1$ if the l-th label is relevant to X_i, which is also called positive label, otherwise $Y_{il} = 0$, also known as negative label ($l \in \{1, 2, \ldots, L\}$), where L is the number of labels. All the labels of each Y_i ($i \in \{1, 2, \ldots, N\}$) in the training set are observed under the setting of MLL with full labels. The goal is to train a model $f : \mathcal{X} \longmapsto [0,1]^L$ that outputs the labels of unseen samples in the feature space $\mathcal{X}$.

SPMLL. For SPMLL, there is only one label observed for each Y_i ($i \in \{1, 2, \ldots, N\}$) in the training set. Formally, Y_{il} satisfies the following two conditions. (1) $Y_{il} \in \{1, \emptyset\}$ for all $i \in \{1, 2, \ldots, N\}$ and $l \in \{1, 2, \ldots, L\}$, where $Y_{il} = \emptyset$ denotes that the l-th label in the i-th sample is unobserved. (2) $\sum_{l=1}^{L} \mathbb{I}_{[Y_{il}=1]} = 1$ for all $i \in \{1, 2, \ldots, N\}$, where $\mathbb{I}_{[\cdot]}$ is the indicator function, which equals 1 when the proposition in the square brackets holds, and 0 otherwise [17]. Obviously, in the SPMLL, the least annotation of each sample makes the supervised information severely insufficient, which causes a performance gap between the SPMLL and MLL with full labels.

3.2 Proposed OPML loss

To close such a performance gap, existing SPMLL methods [17, 18, 23, 24] have made great efforts on it. Most of them assume the unknown labels are negative labels, which inevitably introduces false negatives as noisy labels. More seriously, as the commonly used loss functions in these studies, BCE loss is notoriously not robust to noisy labels [25–27]. To suppress the domination of negative labels, which is the main culprit of fitting noisy labels in SPMLL, we tailor an objective function to push only one pair of labels apart each time. Concretely, we select the observed single positive label and the unobserved label with the maximum score as the pair of labels, and then maximize the difference of the scores between the selected pair of labels to distinguish them. Formally, it can be formulated as follows:

$$\max_{\theta} \left(s_p - \max_{n \in \Omega_n} s_n \right), \tag{1}$$

where θ is the parameter of deep neural networks, s_p and s_n are the scores of a single positive label and assumed negative labels, respectively, and Ω_n is the index set of assumed negative labels. Here, scores are the logits of the neural network; specifically, they refer to unnormalized outputs (without sigmoid activation) of the last layer. Note that the maximum function is non-differentiable; thus we employ the smooth approximation of maximum function logsumexp [45], which is defined as $\text{logsumexp}(x_1, x_2, \ldots, x_k) = \log \sum_{i=1}^{k} e^{(x_i)}$ for any $x_i \in (-\infty, \infty)$ and $i = 1, 2, \ldots, k$, where $\log(\cdot)$ is the natural logarithmic function, and $e^{(\cdot)}$ is the exponential function.

By the smooth approximation of logsumexp and some simple mathematical computations, we have the following objective function:

$$\min_{\theta} \left(-s_p + \log \sum_{n \in \Omega_n} e^{s_n} \right). \tag{2}$$

A detailed derivation from (1) and (2) is provided in Appendix A.

Let $\mathcal{L} = -s_p + \log \sum_{n \in \Omega_n} e^{s_n} = \log\ e^{(-s_p)} + \log \sum_{n \in \Omega_n} e^{s_n}$ be the loss function for SPMLL. Note that this loss is unbounded and the optimal tends to negative infinity, which is unstable and difficult to optimize in deep neural networks. To make this loss function bounded and stable, we add two positive constants α and β into $\mathcal{L}$ and achieve the final loss function customized for SPMLL, defined as $\mathcal{L}_{\text{SPOPML}}$:

$$\mathcal{L}_{\text{SPOPML}} = \log(\alpha + e^{(-s_p)}) + \log \left(\beta + \sum_{n \in \Omega_n} e^{s_n} \right). \tag{3}$$

Note that the objective function in (1) can be easily extended to MLL with full labels by selecting the negative label with the maximum score and the positive label with the minimum score as one pair of

labels to be optimized. Similarly, the formal formulation can be written as

$$\max_{\theta}\left(\min_{p\in\Omega_p}\ s_p-\max_{n\in\Omega_n}\ s_n\right), \tag{4}$$

where Ω_p and Ω_n are the index sets of positive and negative labels, respectively.

For conciseness, similar to the derivation from (1) to (3), we achieve the final loss function for MLL with full labels. The details of deriving (5) are provided in Appendix A.

$$\mathcal{L}_{\mathrm{OPML}}=\log\left(\alpha+\sum_{p\in\Omega_p}e^{(-s_p)}\right)+\log\left(\beta+\sum_{n\in\Omega_n}e^{s_n}\right). \tag{5}$$

It is worth emphasizing that Eq. (5) degenerates to (3) when only a single positive label is observed. Hence, by (5), a unified loss for both SPMLL and MLL with full labels is derived.

Since α and β belong to $(0,\infty)$, which are too wide to select appropriate parameters, here we provide a simple yet flexible mechanism to choose them. Specifically, let $\alpha=\widetilde{\alpha}/(1-\widetilde{\alpha})$ and $\beta=\widetilde{\beta}/(1-\widetilde{\beta})$, where $\widetilde{\alpha}$ and $\widetilde{\beta}$ belong to $(0,1)$. By employing this transformation, we are able to choose the parameters from $(0,1)$ while ensuring that α and β still belong to $(0,\infty)$.

3.3 Gradient analysis

In this subsection, to better understand why BCE loss fits the noisy labels and why our OPML loss performs more robustly than BCE loss in SPMLL, we make detailed gradient analyses for both of them. Besides, we also give a guideline for tuning α and β.

For self-containment, the BCE loss is listed as

$$\mathcal{L}_{\mathrm{BCE}}=-\frac{1}{L}\sum_{l=1}^{L}\left[\mathbb{I}_{[Y_{il}=1]}\log\left(g_{il}\right)\ +\mathbb{I}_{[Y_{il}=0]}\log\left(1-g_{il}\right)\right], \tag{6}$$

where Y_{il} is the l-th label of the i-th sample, and g_{il} is the corresponding predicted score. For the brevity of notation, let $f=f_{il}$, since the sigmoid function $g=1/(1+e^{(-f)})$ is used as the activation function, then the gradient of BCE loss with respect to the logit f can be written as

$$\begin{cases}\frac{\partial\mathcal{L}_{\mathrm{BCE}}}{\partial f}=\frac{\partial\mathcal{L}_{\mathrm{BCE}}}{\partial g}\frac{\partial g}{\partial f}=\frac{-e^{-f}}{1+e^{-f}}, & Y_{il}=1,\\ \frac{\partial\mathcal{L}_{\mathrm{BCE}}}{\partial f}=\frac{\partial\mathcal{L}_{\mathrm{BCE}}}{\partial g}\frac{\partial g}{\partial f}=\frac{e^{f}}{1+e^{f}}, & Y_{il}=0.\end{cases} \tag{7}$$

Note that this gradient is a centrosymmetric function about the origin; thus BCE loss utilizes the same gradient regime for positive and negative labels, which means it treats each label equally [29]. Consequently, BCE loss will fit not only the clean labels but also the noisy labels.

By taking the derivative of (5), the gradients of OPML loss with respect to (w.r.t.) the scores s_p and s_n in the SPMLL setting can be calculated as

$$\begin{cases}\frac{\partial\mathcal{L}_{\mathrm{OPML}}}{\partial s_p}=\frac{-e^{-s_p}}{\alpha+\sum_{p\in\Omega_p}e^{(-s_p)}}=\frac{-e^{-s_p}}{\alpha+e^{(-s_p)}}, & Y_{il}=1,\\ \frac{\partial\mathcal{L}_{\mathrm{OPML}}}{\partial s_n}=\frac{e^{s_n}}{\beta+\sum_{n\in\Omega_n}e^{s_n}}, & Y_{il}=0.\end{cases} \tag{8}$$

In SPMLL, when handling a dataset with severely missing labels, the importance of the positive label needs to be more emphasized; thus relatively small α and large β should be preferred to increase the gradient (absolute value) w.r.t. s_p and decrease the gradient w.r.t. s_n. α and β are not sensitive to datasets with lightly missing labels. Experiments in Subsection 4.4 will also confirm the above analyses.

For ease of comparison, let $\alpha=1$ and $\beta=1$; then we can see that, the gradient of OPML loss w.r.t. the score of the single positive label is the same as that of BCE, while the gradient of OPML loss w.r.t. the score of the negative labels is smaller than that of BCE due to the fact that $e^f/(1+e^f)>e^f/(1+\sum e^f)$. Thus, compared with BCE, the domination of negative labels is suppressed and meanwhile the importance of the positive label is consolidated in SPMLL by adopting the OPML loss for training.

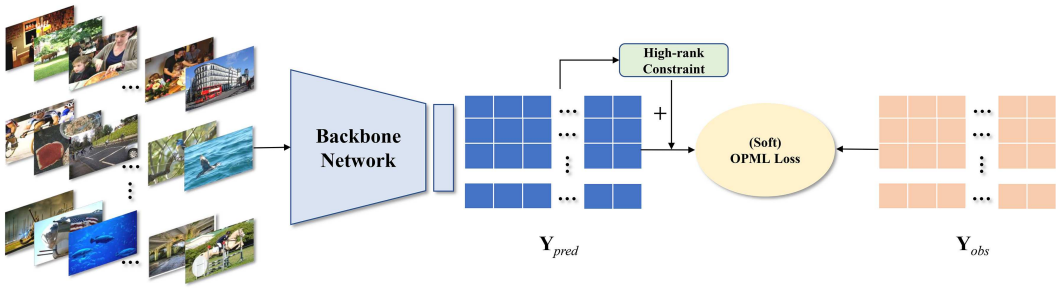

Figure 2 (Color online) Framework of cooperating the high-rank regularization with our OPML loss, where $\boldsymbol{Y}_{\text{pred}}$ and $\boldsymbol{Y}_{\text{obs}}$ denote the predicted and observed label matrices, respectively.

3.4 High-rank regularization

As mentioned in Section 1, in SPMLL, the unknown true positive label will gradually turn to false negative during training. Motivated by the intuition that samples with different labels fall into different subspaces, the label matrix is often prone to be high-rank [31]. To further reduce the risk brought by noisy labels, we explore such high-rank properties to push apart different labels, which aims at further slowing down the process of turning the unknown true positive into a false negative. Specifically, we add a high-rank constraint on the prediction label matrix, which can be formulated as follows:

$$\mathcal{R}_{\text{HR}} = -\lambda \log\det\left(\boldsymbol{Y}_{\text{pred}}^{\text{T}}\boldsymbol{Y}_{\text{pred}}\right) = -\lambda\sum_{i}\log(\sigma_i^2), \tag{9}$$

where λ is a trade-off hyper-parameter, det is the determinant function, $\boldsymbol{Y}_{\text{pred}}$ is the predicted label matrix, and σ_i is its corresponding singular value. Here, we adopt the minus logdet function [46] as the high-rank regularization rather than the minus trace norm used in [31] for the reason that the logdet function is differentiable, which is easier to optimize in deep neural networks. The framework of cooperating this high-rank regularization with our newly proposed OPML loss is depicted in Figure 2.

3.5 Soft variant and label correction

As the OPML loss is customized for SPMLL, a setting with unavoidable noisy labels under the AN assumption, then it is natural to ask whether our OPML loss can be combined with some commonly used techniques in noisy label learning, such as label smoothing [47–49] and label correction [50–52]. To answer this question, in this subsection, we propose a soft variant of OPML loss by label smoothing and further combine it with a label correction mechanism to verify that our loss can be well cooperated with these techniques.

Soft OPML loss. In SPMLL, most methods assume the unknown labels as negatives, which incorrectly annotates the unobserved positives as false negatives. To combat such noisy labels, we utilize the label smoothing to soften the hard discrete labels $\{0, 1\}$ to continuous labels $[0, 1]$. Formally, the soft variant of OPML loss can be rewritten as

$$\mathcal{L}_{\text{SOPML}} = \log(\alpha + e^{(-s_p)}) + \log\left(\alpha + \sum_{l\in\mathcal{U}}\gamma_l e^{(-s_l)}\right) + \log\left(\beta + \sum_{l\in\mathcal{U}}(1-\gamma_l)\, e^{s_l}\right), \tag{10}$$

where $\mathcal{U}$ is the index set of unobserved labels, γ_l is a smoothing parameter of the l-th label. It is worth noting that instead of manual selection, γ_l is a dynamic adaptive parameter computed by the metric of average precision (AP) [53] on the training set. Concretely, $\gamma_l = \text{pred}_l \times \text{AP}_l^{\epsilon}$, where ϵ is a power parameter, pred_l and AP_l are the prediction score and AP score of the l-th label. The intuition of calculating the adaptive parameter in such a way is that the larger the AP, the more reliable its corresponding prediction score is.

Label correction. In noisy label learning, label correction [50–52] is a common and important data cleansing technique. To further validate that our loss can also be well combined with the label correction technique, we likewise propose an AP-based label correction mechanism. Our motivation is intuitive that the larger the AP, the less number the corresponding label is modified. For each class of label, the number of labels to be corrected can be calculated as $\text{Cornum}_l = \text{Tr}_{\text{num}} \times \text{Cor}_{\text{ratio}} \times (1 - \text{AP}_l)$, where Cornum_l is

Algorithm 1 Process of label correction.

Input: Number of training samples $\mathrm{Tr}_{\mathrm{num}}$, label correction ratio $\mathrm{Cor}_{\mathrm{ratio}}$, training average precision score vector AP.
Output: The modified label matrix $\boldsymbol{Y}_m$.

1: **for** $l = 1$ to L **do**
2: Computing the number to be corrected, $\mathrm{Cornum}_l = \mathrm{Tr}_{\mathrm{num}} \times \mathrm{Cor}_{\mathrm{ratio}} \times (1 - \mathrm{AP}_l)$;
3: Sorting the score of unobserved label Score_u in descending order;
4: Changing the negative labels of the first Cornum_l-th samples to positives;
5: **end for**
6: **return** the modified label matrix $\boldsymbol{Y}_m$.

Table 1 Statistics of the datasets.

Dataset	#Class	#Training	#Validation	#Test
CUB	312	4795	1199	5794
NUS	81	120000	30000	60260
COCO	80	65665	16416	40137
VOC	20	4574	1143	5823

the number of labels to be corrected for the l-th label, $\mathrm{Tr}_{\mathrm{num}}$ is the number of training samples, $\mathrm{Cor}_{\mathrm{ratio}}$ is a parameter of label correction ratio, AP_l is the AP score of the l-th label. Note that we use the AP scores of the observed labels in our label smoothing and correction mechanisms. The process of the label correction is summarized in the following algorithm.

The $\mathrm{Cor}_{\mathrm{ratio}}$ in step 2 is computed by $\mathrm{Cor}_{\mathrm{ratio}} = \mathrm{Obsnum}_l/\mathrm{Tr}_{\mathrm{num}} \times \mathrm{Label}_{\mathrm{num}}$, where Obsnum_l is the number of observed labels in the l-th label, and $\mathrm{Label}_{\mathrm{num}}$ is a parameter that will be specified in Subsection 4.8.

In summary, to cooperate the OPML loss with the label correction, label smoothing, and the high-rank constraint, we first preprocess the label correction by running algorithm 1, then use (10) as the loss function, and plus (9) as the regularization term. Finally, the overall objective function can be written as

$$\mathcal{L}_O = \mathcal{L}_{\mathrm{SOPML}} + \mathcal{R}_{\mathrm{HR}}. \tag{11}$$

4 Experiments

In this section, we conduct extensive experiments on multi-label image classification in both single positive and full labels settings to verify the effectiveness of our proposed method.

4.1 Experiment settings

Datasets. For fairness of comparisons, we follow the setting in [17, 23, 29]. Specifically, four standard benchmark datasets, PASCAL VOC 2012 (VOC) [54], MS-COCO 2014 (COCO) [55], NUS-WIDE (NUS) [56], and CUB-200-2011 (CUB) [57] are used for evaluation. Given four full labels datasets, first, for each dataset, 20% of the training set is withheld for validation. Second, to create single positive training data, one positive label is randomly selected to keep for each training sample, the remaining annotations are discarded, which is performed only once for each dataset, and the single positive training set is fixed. The statistics of four datasets are shown in Table 1. Note that we use totally the same number of seeds as in [17, 23, 29] for random sampling and spitting to create the same dataset for fair comparisons.

Compared methods. For SPMLL, we compare our method with two kinds of methods: one is the methods customized for SPMLL, and the other is the commonly used loss functions in MLL. The latter contains BCE loss, focal loss [37], asymmetric loss [36], and ZLPR loss [42], and the former includes BCE + DW (down-weighting negative labels), BCE + L1R/L2R (l_1/l_2 regularization), BCE + LS (label smoothing), BCE + N-LS (label smoothing for only negative labels), ROLE (regularized online label estimation) [17], ROLE + LI (ROLE combined with the "LinearInit") [17], LL-R (large loss with rejection) [23], LL-Ct (large loss with temporary correction) [23], LL-Cp (large loss with permanent correction) [23], BCE + EntMax (entropy maximization regularization) [29], and BCE + EntMax + APL (BCE + entropy maximization regularization + asymmetric pseudo-labeling) [29].

Implementation details. Following [17, 23, 29], ResNet-50 [58] pretrained on the ImageNet [59] is adopted as the backbone network in all the experiments. Although simply using the up-to-date weights

Table 2 Compared results of the methods customized for SPMLL and our methods on four SPMLL benchmarks. The mean and standard deviation of mAP are reported. 1 P. & All N. means that one positive and all the negative labels are observed, and 1 P. & 0 N. signifies that only one positive label is observed while others remain unknown, i.e., the single positive setting. The best results are in bold and the second best results are underlined.

Observed label	Method	CUB	NUS	COCO	VOC
Full labels	BCE	30.90(0.64)	52.08(0.20)	76.78(0.13)	89.42(0.27)
1 P. & All N.	BCE	20.65(1.11)	46.45(0.27)	71.39(0.19)	87.60(0.31)
1 P. & 0 N.	BCE	18.31(0.47)	42.27(0.56)	64.92(0.19)	85.89(0.38)
	BCE+DW	19.15(0.56)	45.71(0.23)	67.59(0.11)	86.98(0.36)
	BCE+L1R	17.59(1.82)	42.15(0.46)	64.44(0.20)	85.97(0.31)
	BCE+L2R	17.71(1.79)	42.72(0.12)	64.41(0.24)	85.96(0.36)
	BCE+LS	16.26(0.45)	43.77(0.29)	67.15(0.13)	87.90(0.21)
	BCE+N-LS	16.82(0.42)	43.86(0.54)	67.15(0.10)	88.12(0.32)
	ROLE (CVPR21)	13.66(0.24)	41.63(0.35)	67.04(0.19)	87.77(0.22)
	ROLE+LI (CVPR21)	14.86(0.72)	45.98(0.26)	69.12(0.13)	88.26(0.21)
	LL-R (CVPR22)	20.55(0.18)	48.10(0.12)	70.36(0.21)	88.79(0.03)
	LL-Ct (CVPR22)	20.53(0.18)	<u>48.18(0.17)</u>	70.27(0.08)	88.80(0.11)
	LL-Cp (CVPR22)	20.55(0.22)	47.92(0.02)	70.37(0.13)	88.37(0.23)
	BCE+EntMax (ECCV22)	20.85(0.42)	47.15(0.11)	70.70(0.31)	89.09(0.17)
	BCE+EntMax+APL (ECCV22)	21.84(0.34)	47.59(0.22)	<u>70.87(0.23)</u>	<u>89.19(0.31)</u>
1 P. & 0 N.	OPML (ours)	<u>22.30(0.08)</u>	47.93(0.05)	68.93(0.06)	87.77(0.04)
	OPML-SP (ours)	**24.11(0.22)**	**50.14(0.09)**	**71.75(0.07)**	**89.20(0.03)**

of ResNet-50 pretrained on the ImageNet can achieve better performance, we still use the old V1 version as the compared method for fair comparisons. The grid search is adopted for model selection, and the best hyper-parameters are selected by the best mean average precision (mAP) on the validation set. Each experiment runs three times, and both the mean and standard deviation of mAP are reported. For reproducing, the best parameters for each dataset are provided in Subsection 4.8.

4.2 Experimental results of SPMLL

In this subsection, we report the results of our methods and the compared methods by conducting experiments on four SPMLL benchmarks, and then make detailed analyses.

In Table 2, BCE loss in the SPMLL setting can be seen as a baseline, and methods starting from "ROLE" focus on SPMLL. We also report the results of BCE loss with full labels as an oracle. Our method OPML-SP is short for OPML loss for a single positive setting, which is the soft OPML loss with high-rank regularization and label correction. From Table 2, we have the following findings.

(1) Compared with full labels, the mAP of BCE in SPMLL drops dramatically, e.g., 11.86% and 12.59% decrease on COCO and CUB, respectively, which illustrates that BCE loss is not appropriate for SPMLL.

(2) Although some of the BCE variants improve the performance, there still exists a large gap between the SPMLL and full labels, especially on CUB, which severely lacks labels in SPMLL. Besides, the second best results are not obtained by the same method, which indicates that the compared methods are not suitable for all the four datasets. Whereas our OPML-SP performs the best on all four datasets and closes the performance gap between SPMLL and full labels, e.g., there are only 0.22% and 1.94% differences on VOC and NUS, respectively. And on the most difficult dataset CUB, OPML-SP achieves a new state-of-the-art result, which demonstrates that our method performs more robustly than the compared methods even when there severely lacks labels.

(3) Most surprisingly, even in the SPMLL setting, OPML-SP still performs better than BCE in the one positive and all negative labels observed setting. Besides, only adopting the OPML loss defeats state-of-the-art SPMLL methods with regularizations or fine-tuned label correction on CUB. These two funny results can be attributed to the fact that our strategy of pushing one pair of labels apart each time not only prevents the domination of negative labels but also consolidates the importance of the observed single positive label.

Remark 1. Note that in Table 2, the mAP of our method on VOC in SPMLL (89.20) is very close to the full label setting (89.42); thus a further improvement is rather limited. For COCO, the real mAP should be higher than what we report, since some missing ground truth labels, e.g., car and book indeed appear, and our method can discover them. For details, please refer to Subsection 4.6.

Table 3 Ranking loss (↓) of the methods customized for SPMLL and our OPML loss on four SPMLL benchmarks. The mean and standard deviation of ranking loss are reported. The best results are in bold and the second best results are underlined.

Method	CUB	NUS	COCO	VOC
BCE	0.166(0.006)	0.015(0.000)	0.025(0.001)	0.014(0.001)
ROLE	0.305(0.008)	0.025(0.001)	0.036(0.001)	0.017(0.001)
LL	0.165(0.001)	0.014(0.001)	0.022(0.000)	0.012(0.001)
APL	0.159(0.006)	0.018(0.001)	0.026(0.001)	0.013(0.001)
OPML	**0.140(0.001)**	**0.013(0.000)**	**0.020(0.001)**	**0.011(0.000)**

Table 4 One error (↓) of the methods customized for SPMLL and our OPML loss on four SPMLL benchmarks. The mean and standard deviation of one error are reported. The best results are in bold and the second best results are underlined.

Methods	CUB	NUS	COCO	VOC
BCE	0.380(0.045)	0.374(0.002)	0.088(0.002)	0.059(0.004)
ROLE	0.293(0.058)	0.371(0.002)	.066(0.002)	0.050(0.002)
LL	0.345(0.146)	0.361(0.001)	0.057(0.003)	0.055(0.003)
APL	0.173(0.014)	0.360(0.000)	0.055(0.002)	0.045(0.001)
OPML	**0.157(0.004)**	**0.359(0.001)**	**0.054(0.001)**	**0.044(0.001)**

Table 5 Quantitative results of the commonly used loss functions and our OPML loss on four SPMLL benchmarks. The mean and standard deviation of mAP are reported. The best results are in bold and the second best results are underlined.

Methods	CUB	NUS	COCO	VOC
BCE	18.31(0.47)	42.27(0.56)	64.92(0.19)	85.89(0.38)
FOCAL	19.80(0.30)	47.00(0.14)	68.79(0.14)	87.59(0.58)
ASL	18.81(0.48)	46.93(0.30)	68.78(0.32)	87.76(0.51)
ZLPR	21.02(0.12)	47.72(0.14)	68.41(0.24)	87.63(0.03)
OPML	**22.30(0.08)**	**47.93(0.05)**	**68.93(0.06)**	**87.77(0.04)**

We also conduct experiments on the metrics of ranking loss and one error; please refer to [53] for the specific definitions of these two metrics. The results of the two metrics are presented in Tables 3 and 4, respectively. For both the ranking loss and one error, the results are the lower the better. From the above two tables, we can conclude that our OPML loss consistently obtains the best results on all the four SPMLL benchmarks. The ranking loss reflects the error of incorrectly ordered labels between the positive and negative labels. As the OPML loss effectively alleviates the imbalance between the positive and negative labels, which helps us learn a better ranking order for the positive and negative labels. Consequently, our OPML loss achieves good performance with respect to the ranking loss. Note that the metric of one error primarily focuses on the Top-1 prediction accuracy, which may be a relatively loose metric for multi-label learning. Therefore, the advantage of the performance on some datasets may be minor. However, the advantage on the CUB dataset is considerable for both metrics, which again demonstrates the superiority of our OPML loss in dealing with more challenging datasets where labels are heavily scarce.

In Table 5, results of the commonly used loss functions and our OPML loss are displayed. From Table 5, we can find that the performance of BCE loss is the worst, which also verifies that BCE loss is not appropriate for SPMLL. Although the variants of BCE loss, i.e., focal and asymmetric losses improve the performance compared with BCE loss, they are still not good enough. Moreover, on the dataset CUB, the improvements are marginal; the reason for this result is that in the setting of SPMLL, CUB suffers from a severe lack of labels, making it more susceptible to noisy labels. Besides, as a special case of our OPML loss, ZLPR achieves an inferior performance, which validates that OPML loss is more flexible and has more potential in SPMLL.

In the following, we conduct statistical experiments to see the distribution of predicted confidence with respect to the positive labels for the test set. First, we find the positions where the ground truth in the test label matrix are positive labels, and then correspondingly, the predicted confidence scores at the same position in the predicted label matrix are selected for statistical experiments. Figure 3 shows the statistical histogram of the predicted confidence corresponding to where the ground truth is positive labels. Thus the more the predicted confidence falls into $[0.8, 1]$, the more accurate the prediction is. From Figure 3, we can find that in the interval of $[0.8, 1]$, the frequency of OPML is larger than BCE,

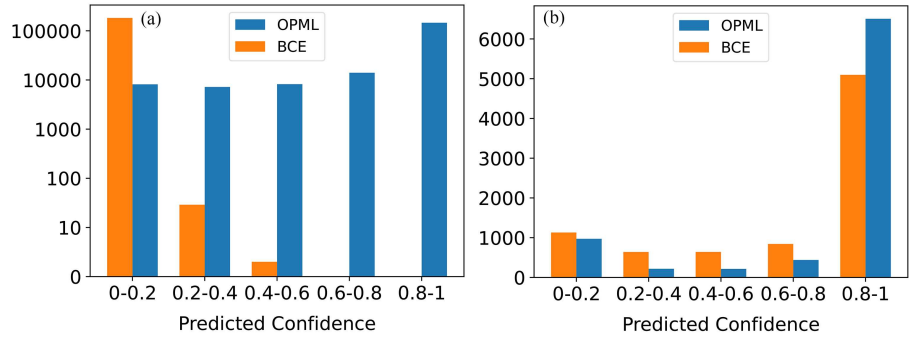

Figure 3 Histograms of the predicted confidence corresponding to the positive labels in the test label matrix. The "OPML" with blue and "BCE" with orange denote the histogram of the predicted confidence trained with OPML and BCE loss, respectively. The width of each bin is 0.2, noting that in Figure 3(a), when trained with BCE loss, the frequencies of the predicted confidence that falls into the interval of $[0.6, 0.8)$ and $[0.8, 1]$ are zero. (a) Histogram on CUB; (b) histogram on VOC.

which means when trained with the OPML loss, the predicted confidence scores of positive labels are more inclined to 1. Besides, in the interval of $[0, 0.2]$, the frequency of OPML is smaller than BCE, indicating that OPML can alleviate the domination of negative labels in SPMLL. This phenomenon is more obvious in Figure 3(a) that, when trained with BCE loss, the frequencies of the predicted confidence that falls into the interval of $[0.6, 0.8]$ and $[0.8, 1]$ are zero, which means that compared with BCE loss, our OPML loss has a more prominent advantage when the dataset is with more unannotated positive labels.

4.3 Ablation study

In this subsection, we conduct experiments to verify the effectiveness of the high-rank regularization. Besides, we also validate that our OPML loss can be well cooperated with label smoothing and label correction, which are commonly used techniques in noisy label learning. Unless otherwise specified, the experiments in Subsections 4.2–4.4, and 4.6 are all performed in the setting of a single positive.

High-rank regularization. Compared with the first row in Table 6, we can find that the mAP scores of the second row on all four datasets increase, which verifies the effectiveness of high-rank regularization. Besides, to validate the performance drop is not that dramatic with the high-rank regularization, we conduct experiments adopting BCE loss, OPML loss, and OPML loss with high-rank regularization, respectively. Results of running 40 epochs on CUB and VOC are shown in Figure 4. It can be seen that the mAP scores are more stable with the high-rank regularization. The reason may be that the high-rank property encourages to push different labels apart, which slows the process of turning the true positive labels to false negative labels. Thus, such a property reduces the impact of noisy labels and achieves a stable training in SPMLL. We hope this discovery can bring new inspiration to general noisy label learning.

Label smoothing and correction. From the third and the fourth rows of Table 6, we can find that the performance is improved on all the four datasets with both the label smoothing and correction. Improvement on CUB is less than the other three datasets with label smoothing, which can be attributed to the fact that the ground truth of CUB contains more positive labels, while label smoothing tries to transform the hard labels $\{0, 1\}$ to continuous $[0, 1]$, such an operation does not pay enough attention to the positive labels. Besides, label correction obtains noticeable improvement on all the four datasets, which validates that our training AP-based correction strategy is effective in SPMLL. To sum up, all the experiments illustrate that our OPML loss can be well cooperated with the label smoothing and correction.

4.4 Hyper-parameter study

In this subsection, we conduct experiments with different values of hyper-parameters to study their effects. Note that the best hyper-parameter is selected by the best mAP on the validation set. In Figure 5, we show the results with different hyper-parameters of α and β in (5). Recalling that we transform $\alpha = \widetilde{\alpha}/(1-\widetilde{\alpha})$ and $\beta = \widetilde{\beta}/(1-\widetilde{\beta})$ for convenience of hyper-parameters selection, thus results are reported with different $\widetilde{\alpha}$ and $\widetilde{\beta}$. The best hyper-parameter is denoted by the orange pillar. For example, $\widetilde{\alpha} = 0.9$ and $\widetilde{\beta} = 0.5$

Table 6 Results of ablation study on four SPMLL benchmarks. The ✓ means with the corresponding component. The mean and standard deviation of mAP are reported, and the best results are in bold.

$\mathcal{L}_{\text{OPML}}$	$\mathcal{R}_{\text{HR}}$	Smoothing	Correction	CUB	NUS	COCO	VOC
✓	–	–	–	22.30(0.08)	47.93(0.05)	68.93(0.06)	87.77(0.04)
✓	✓	–	–	22.41(0.10)	48.11(0.06)	69.36(0.02)	87.81(0.11)
✓	–	✓	–	22.37(0.06)	48.98(0.20)	70.42(0.13)	88.62(0.13)
✓	–	–	✓	23.62(0.18)	48.97(0.07)	69.38(0.18)	88.36(0.10)
✓	✓	✓	✓	**24.11(0.22)**	**50.14(0.09)**	**71.75(0.07)**	**89.20(0.03)**

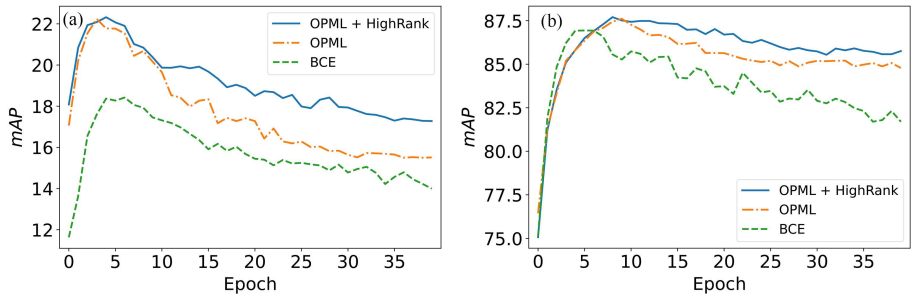

Figure 4 Effectiveness of high-rank regularization. The performance drop is not that dramatic with such a regularization. (a) mAP of 40 epochs on CUB; (b) mAP of 40 epochs on VOC.

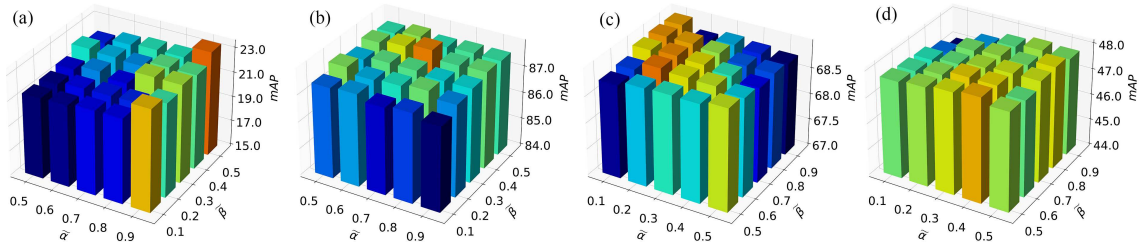

Figure 5 Hyper-parameter study of $\widetilde{\alpha}$ and $\widetilde{\beta}$. The best performance is marked with orange. mAP with different $\widetilde{\alpha}$ on (a) CUB, (b) VOC, (c) COCO, and (d) NUS.

are the best parameters for CUB, and $\widetilde{\alpha} = 0.7$ and $\widetilde{\beta} = 0.4$ are the best parameters for VOC. Note that, as a special case of our loss, the mAP values of ZLPR ($\widetilde{\alpha} = 0.5$ and $\widetilde{\beta} = 0.5$) are lower than our OPML loss, which indicates that OPML is more flexible and achieves better performance.

Besides, we also carry out experiments to study the sensitivity of the hyper-parameter λ in (9). To better observe the downtrend when λ is larger than $1\text{E}-2$, we add the mAP values of $\lambda = 2\text{E}-2$ and $\lambda = 5\text{E}-2$. From Figure 6, we can see that the mAP value keeps stable when λ is below $1\text{E}-2$. The best λ for CUB and NUS is $1\text{E}-3$, and the best λ for VOC and COCO is $1\text{E}-2$. These figures demonstrate that λ is not sensitive to variations when less than $1\text{E}-2$.

4.5 Convergence and running time study

In this subsection, to verify that our OPML does not slow down the learning procedure, we study the convergence and running time of the OPML loss. Since the framework used in this paper is a deep neural network (DNN), it is well known that the rigorous convergence proof of DNN remains an open problem so far; therefore we provide the curve of the loss function in Figure 7 as an alternative to describe its convergence.

Furthermore, we also provide the computational complexity analysis and the running time for 1 epoch. The computation complexity of the gradient update can be obtained by analyzing 8. The complexity of updating the positive score is $O(B \times 1)$, where B is the size of mini-batch, and the complexity of updating the negative scores is $O(B \times (L-1))$, where L is the number of all labels; thus the total computation complexity of the gradient update is $O(B \times L)$. Besides, we report the running time in both SPMLL and full label settings. Note that, ROLE [17], LL [23], and APL [29] are methods customized for SPMLL, while FOCAL [37], ASL [36], and ZLPR [42] are commonly used loss functions in MLL. From the results in Table 7, we can see that our OPML loss does not introduce any extra overhead, and the running

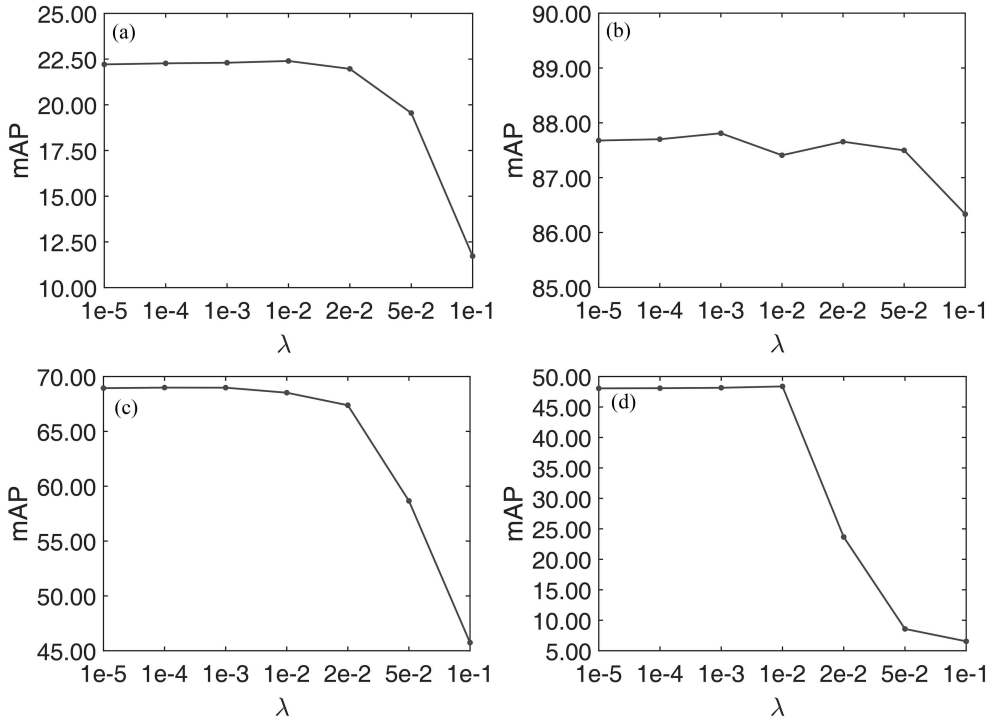

Figure 6 Sensitivity study of λ. It is not sensitive to variations when less than $1E-2$ on both datasets. mAP with different λ on (a) CUB, (b) VOC, (c) COCO, and (d) NUS.

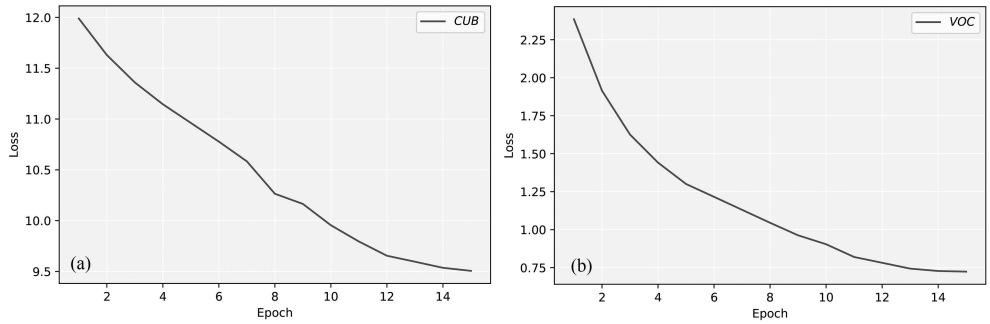

Figure 7 Training losses on the (a) CUB and (b) VOC datasets.

Table 7 Running time for 1 epoch of each method.

SPMLL setting					Full labels setting				
Method	CUB	NUS	COCO	VOC	Method	CUB	NUS	COCO	VOC
BCE	1 m 28 s	29 m 43 s	16 m 17 s	1 m 14 s	BCE	1 m 17s	29 m 46 s	16 m 20 s	1 m 14 s
ROLE	1 m 15 s	29 m 36 s	16 m 12 s	1 m 05 s	FOCAL	1 m 18 s	29 m 49 s	16 m 19 s	1 m 15 s
LL	1 m 11 s	29 m 31 s	16 m 06 s	1 m 04 s	ASL	1 m 18 s	29 m 48 s	16 m 20 s	1 m 15 s
APL	1 m 22 s	30 m 06 s	16 m 20 s	1 m 10 s	ZLPR	1 m 18 s	29 m 47 s	16 m 19 s	1 m 15 s
OPML	1 m 14 s	29 m 37 s	16 m 10 s	1 m 03 s	OPML	1 m 18 s	29 m 42 s	16 m 16 s	1 m 14 s

time is almost the same as other compared methods. These results demonstrate that the OPML loss can be seamlessly integrated into existing frameworks without any adverse impact on the computational efficiency.

4.6 Grad-CAM visualization

In this subsection, we utilize the gradient-weighted class activation mapping (Grad-CAM) [60] to visualize the explanation of the proposed OPML loss, which produces a coarse localization map highlighting the important regions in the image for predicting the concept. In Figure 8, we show seven groups of pictures

Table 8 Results of commonly used loss functions in MLL and our OPML loss on four benchmarks with full labels. The mean and standard deviation of mAP are reported. The best results are in bold and the second best results are underlined.

Method	CUB	NUS	COCO	VOC
BCE	30.90(0.64)	52.08(0.20)	76.78(0.13)	89.42(0.27)
FOCAL	33.36(0.05)	53.20(0.08)	<u>77.35(0.13)</u>	90.74(0.07)
ASY	<u>33.37(0.19)</u>	52.84(0.17)	**77.61(0.32)**	90.60(0.36)
ZLPR	33.27(0.12)	<u>53.40(0.15)</u>	76.35(0.10)	<u>90.96(0.05)</u>
OPML	**33.72(0.08)**	**53.98(0.20)**	76.71(0.16)	**91.36(0.06)**

and their class activation maps, respectively. It can be seen that our OPML loss is capable of highlighting the locations of the concept to be recognized. It is worth mentioning that in Figures 8(f) and (g), the class activation maps discover missing concepts that are neglected in the ground truth but actually exist in the image. For example, in Figure 8(f), the ground truth labels ignore the "potted plant" and "car", which indeed appear in the image. Especially for "car", it can hardly be recognized even by humans, and the same phenomenon can also be found in Figure 8(g). These intriguing findings demonstrate that our OPML loss not only has an explanatory power that matches with humans, but also has the potential to help humans complete the missing labels, which can provide more accurate and complete labels for training.

Additionally, we also conduct experiments to draw the class activation mapping of other compared methods, aiming at further conforming the effectiveness of our method. From Figure 9, we have the following findings. (1) For the missing ground truth label "book", all the compared methods cannot accurately locate the position of the "book" whereas our OPML loss can accurately draw the Class Activation Mapping of the label "book", which demonstrates that the OPML loss has more potential in completing the missing ground truth labels. (2) From Figure 9(a), we can find that for the labels "person" and "horse", BCE loss fails to learn them, which can be attributed to the fact that under the single positive setting, BCE loss is easily stuck in the domination of the negative labels during training. (3) Overall, compared with other methods, our OPML loss not only highlights more accurate regions of the class activation mapping, but also has more potential in completing the missing ground truth labels, which shows the superiority of our method. According to gradient analysis in Subsection 3.3, when comparing (7) and (8), we can find that our OPML loss obtains a more balanced gradient update than the BCE loss due to the fact that $e^f/(1+e^f) > e^f/(1+\sum e^f)$. Therefore, OPML loss places greater emphasis on the positive labels and exhibits a higher potential to discover more positive labels, especially for those that are challenging to identify, which may explain why our OPML loss can discover missing concepts that are neglected in the ground truth.

4.7 Experimental results of full labels

In this subsection, we conduct experiments on MLL with full labels to validate that our OPML loss can still work well in such a setting, even though it is originated from the SPMLL setting. Note that ASY [36] is a state-of-the-art method in MLL with full labels, which focuses on addressing the imbalance between positive and negative labels. From Table 8, we can find that our OPML loss performs the best on three datasets, which can be attributed to the fact that pushing the positive label with the minimum score and negative label with the maximum score apart may increase the discrimination between them.

4.8 Details of experimental implementation

To better reproduce the experimental results reported in this work, we elaborate on the details of experimental implementation in this subsection. The stochastic gradient descent (SGD) optimizer is used for training. The parameters of OPML-SP (the abbreviation of OPML loss for single positive setting) on dataset VOC are listed as follows: batch size is 8, learning rate is $1\mathrm{E}-4$, running epochs are 15, $\widetilde{\alpha}=0.6$, $\widetilde{\beta}=0.4$, smoothing power parameter $\epsilon=0.7$, label correction parameter $\mathrm{Label}_{\mathrm{num}}=0.8$. The parameters used on COCO are that batch size is 16, learning rate is $2\mathrm{E}-4$, running epochs are 10, $\widetilde{\alpha}=0.4$, $\widetilde{\beta}=0.5$, smoothing power parameter $\epsilon=1$, and label correction parameter $\mathrm{Label}_{\mathrm{num}}=1.2$. Besides, the parameters used on NUS are that batch size is 128, the learning rate is $1\mathrm{E}-3$, running epochs are 10, $\widetilde{\alpha}=0.6$, $\widetilde{\beta}=0.4$, smoothing power parameter $\epsilon=1$, label correction parameter $\mathrm{Label}_{\mathrm{num}}=1.1$. Finally, the parameters used on CUB are as below: batch size is 8, learning rate is $4\mathrm{E}-4$, running epochs are 10, $\widetilde{\alpha}=0.6$, $\widetilde{\beta}=0.2$, smoothing power parameter $\epsilon=1$, and the label correction parameter $\mathrm{Label}_{\mathrm{num}}=22$.

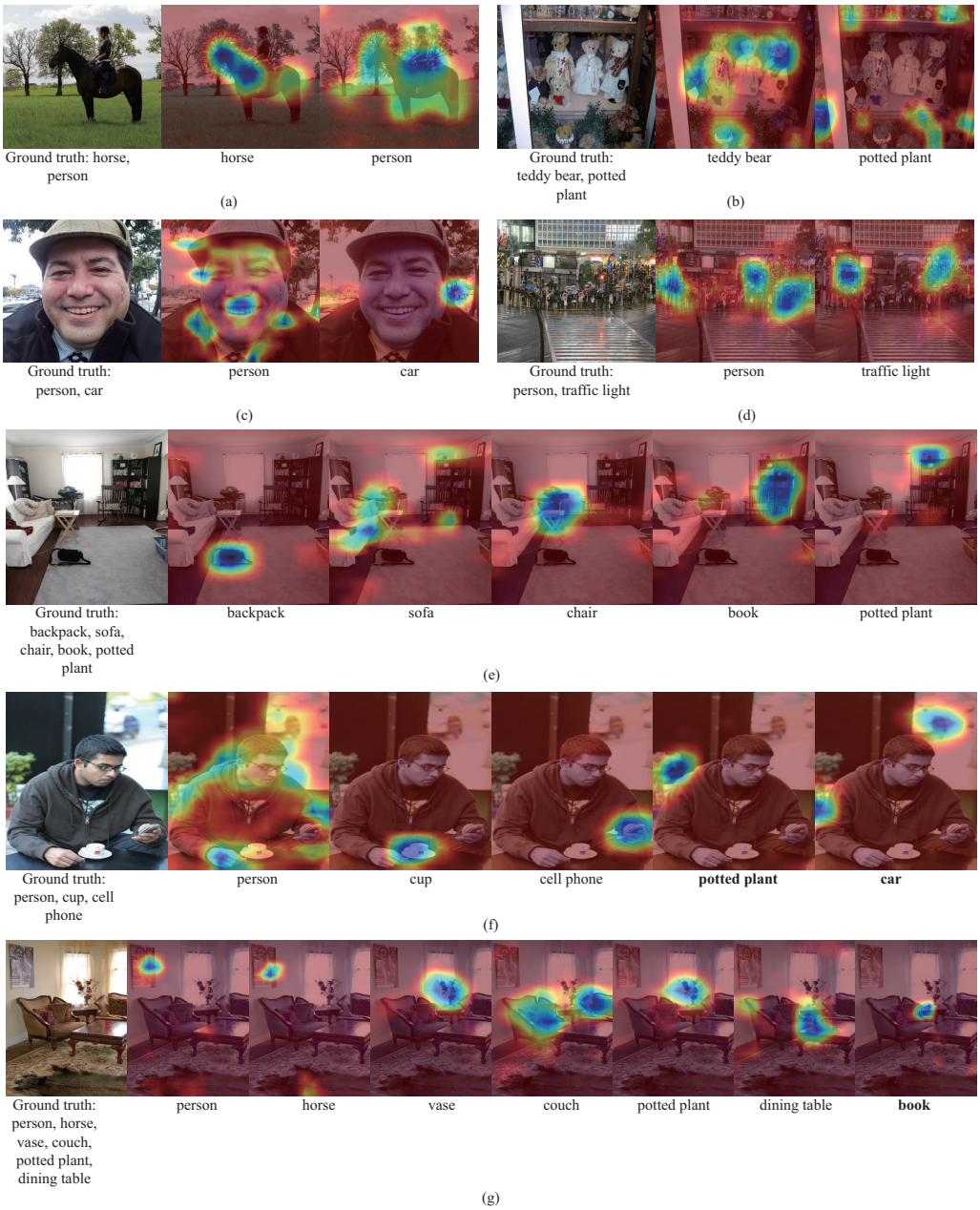

Figure 8 Class activation mapping of COCO test images. The blue color highlights the important regions in the image for predicting the corresponding concept. Labels in bold (potted plant and car in (f) and book in (g)) are the missing ground truth labels that can be discovered by our OPML loss.

The hyper-parameter λ is fixed at $1\mathrm{E}-3$ on all four datasets. Note that the parameters reported here are not fine-tuned, and performance may be further improved with fine-tuned parameters.

5 Conclusion

In this paper, we present a novel unified loss named OPML for both SPMLL and MLL with full labels by pushing one pair of labels apart each time to suppress the domination of negative labels. Experiments on four benchmarks verify that the OPML loss not only performs more robustly in SPMLL for alleviating the impact of noisy labels but also works well in MLL with full labels for separating the positive and negative labels. Besides, we empirically find that the high-rank property of the label matrix can slow down the dramatic performance drop, which may shed new light on general noisy label learning. Note that the imbalance between the positive and negative labels becomes more severe in SPMLL; thus, how

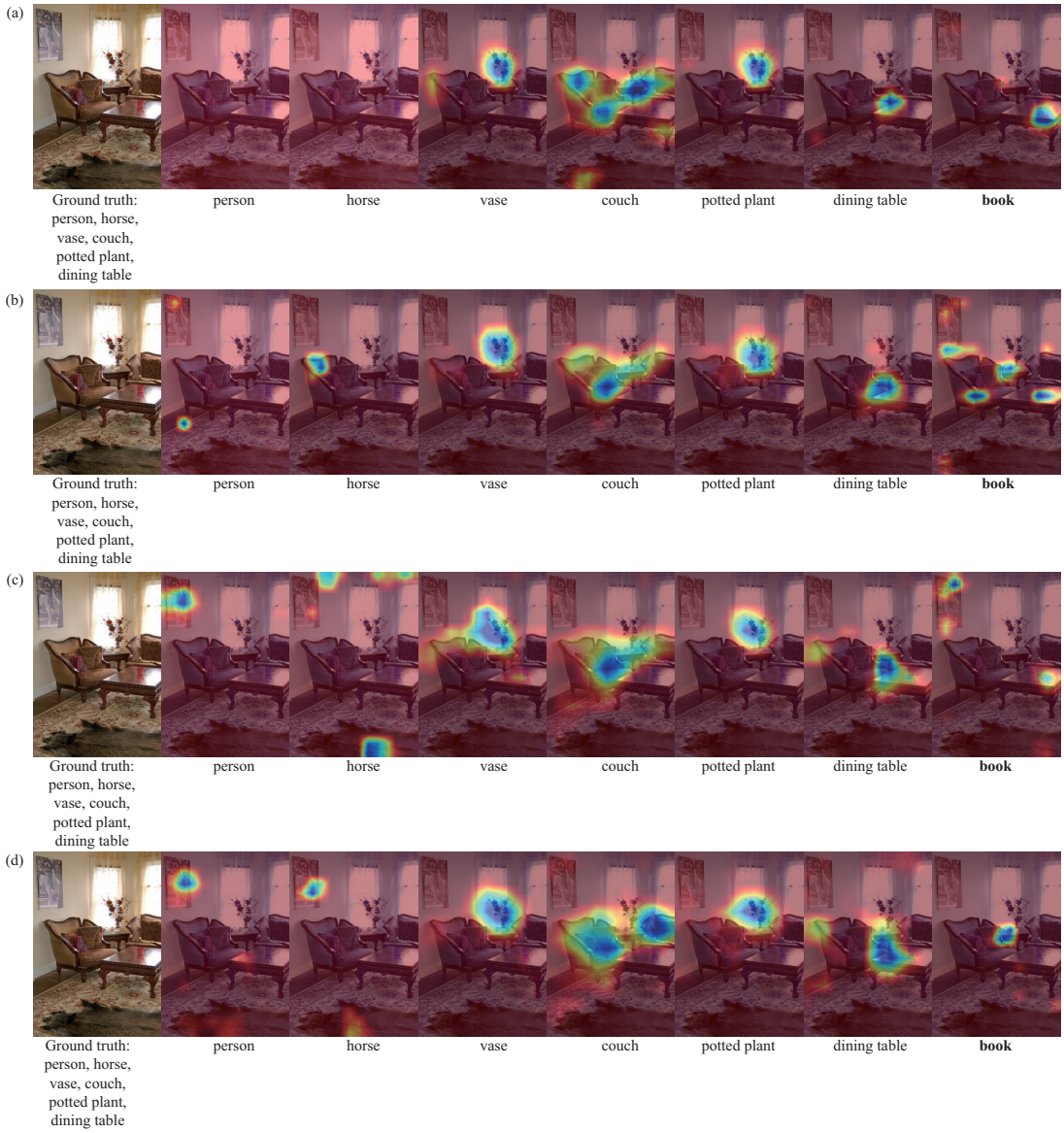

Figure 9 Class activation mapping of COCO test images for different methods. The blue color highlights the important regions in the image for predicting the corresponding concept. Label in bold (book) is the missing ground truth label, which can be accurately discovered by our OPML loss. (a) BCE loss; (b) ROLE loss; (c) large loss; (d) our OPML loss.

to deal with this issue may be a future research direction for further closing the gap between SPMLL and MLL with full labels.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 62376126).

References

1 Zhang M L, Zhou Z H. A review on multi-label learning algorithms. IEEE Trans Knowl Data Eng, 2014, 26: 1819–1837
2 Gibaja E, Ventura S. A tutorial on multilabel learning. ACM Comput Surv, 2015, 47: 1–38
3 Liu W, Wang H, Shen X, et al. The emerging trends of multi-label learning. IEEE Trans Pattern Anal Mach Intell, 2022, 44: 7955–7974
4 Zhuang Y T, Han Y H, Wu F, et al. Stable multi-label boosting for image annotation with structural feature selection. Sci China Inf Sci, 2011, 54: 2508–2521
5 Lanchantin J, Wang T, Ordonez V, et al. General multi-label image classification with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021. 16478–16488
6 Chen M T, Wang X G, Luo H, et al. Learning to focus: cascaded feature matching network for few-shot image recognition. Sci China Inf Sci, 2021, 64: 192105
7 Dong N, Wang J, Voiculescu I. Revisiting vicinal risk minimization for partially supervised multi-label classification under data scarcity. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 4212–4220

8 Cheng G, Lai P J, Gao D C, et al. Class attention network for image recognition. Sci China Inf Sci, 2023, 66: 132105
9 Ray J, Wang H, Tran D, et al. Scenes-objects-actions: a multi-task, multi-label video dataset. In: Proceedings of the European Conference on Computer Vision, 2018. 635–651
10 Zhou W, Liu J, Lei J, et al. GMNet: graded-feature multilabel-learning network for RGB-thermal urban scene semantic segmentation. IEEE Trans Image Process, 2021, 30: 7790–7802
11 Zhang Y, Li X, Marsic I. Multi-label activity recognition using activity-specific features and activity correlations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021. 14625–14635
12 Ji W, Wang R. A multi-instance multi-label dual learning approach for video captioning. ACM Trans Multimedia Comput Commun Appl, 2021, 17: 1–18
13 Habimana O, Li Y H, Li R X, et al. Sentiment analysis using deep learning approaches: an overview. Sci China Inf Sci, 2020, 63: 111102
14 Chen B, Huang X, Xiao L, et al. Hyperbolic interaction model for hierarchical multi-label classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020. 7496–7503
15 Zhang J, Chang W C, Yu H F, et al. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 7267–7280
16 Amio E, Delgado A. Evaluating extreme hierarchical multilabel classification. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022. 5809–5819
17 Cole E, Aodha O M, Lorieul T, et al. Multi-label learning from single positive labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021. 933–942
18 Xu N, Qiao C, Lv J, et al. One positive label is sufficient: single-positive multi-label learning with label enhancement. In: Proceedings of Advances in Neural Information Processing Systems, 2022. 21765–21776
19 Weng W, Lin Y, Wu S, et al. Multi-label learning based on label-specific features and local pairwise label correlation. Neurocomputing, 2018, 273: 385–394
20 Nguyen T T, Nguyen T T T, Luong A V, et al. Multi-label classification via label correlation and first order feature dependance in a data stream. Pattern Recogn, 2019, 90: 35–51
21 Che X, Chen D, Mi J. A novel approach for learning label correlation with application to feature selection of multi-label data. Inf Sci, 2020, 512: 795–812
22 Xu M, Guo L-Z. Learning from group supervision: the impact of supervision deficiency on multi-label learning. Sci China Inf Sci, 2021, 64: 130101
23 Kim Y, Kim J M, Akata Z, et al. Large loss matters in weakly supervised multi-label classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022. 14156–14165
24 Xie M K, Xiao J H, Huang S J. Label-aware global consistency for multi-label learning with single positive labels. In: Proceedings of Advances in Neural Information Processing Systems, 2022. 18430–18441
25 Ma X, Huang H, Wang Y, et al. Normalized loss functions for deep learning with noisy labels. In: Proceedings of International Conference on Machine Learning, 2020. 6543–6553
26 Wang D B, Wen Y, Pan L, et al. Learning from noisy labels with complementary loss functions. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2021. 10111–10119
27 Zhou X, Liu X, Wang C, et al. Learning with noisy labels via sparse regularization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 72–81
28 Zhao W, Gomes C. Evaluating multi-label classifiers with noisy labels. 2021. ArXiv:2102.08427
29 Zhou D, Chen P, Wang Q, et al. Acknowledging the unknown for multi-label learning with single positive labels. In: Proceedings of European Conference on Computer Vision, 2022. 423–440
30 Guermeur Y. VC theory of large margin multi-category classifiers. J Mach Learn Res, 2007, 8: 2551–2594
31 Li X, Chen S. A concise yet effective model for non-aligned incomplete multi-view and missing multi-label learning. IEEE Trans Pattern Anal Mach Intell, 2022, 44: 5918–5932
32 Bartlett P L, Wegkamp M H. Classification with a reject option using a hinge loss. J Mach Learn Res, 2008, 9: 1823–1840
33 Liu Y, Sheng L, Shao J, et al. Multi-label image classification via knowledge distillation from weakly-supervised detection. In: Proceedings of the 26th ACM International Conference on Multimedia, 2018. 700–708
34 Chen Z M, Wei X S, Wang P, et al. Multi-label image recognition with graph convolutional networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019. 5177–5186
35 Gao B B, Zhou H Y. Multi-label image recognition with multiclass attentional regions. 2020. ArXiv:2007.01755
36 Ridnik T, Ben-Baruch E, Zamir N, et al. Asymmetric loss for multi-label classification. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021. 82–91
37 Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2017. 2999–3007
38 Wu T, Huang Q, Liu Z, Wang Y, et al. Distribution-balanced loss for multi-label classification in long-tailed datasets. In: Proceedings of European Conference on Computer Vision, 2020. 162–178
39 Akyürek A F, Guo L, Elanwar R, et al. Multi-label and multilingual news framing analysis. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020. 8614–8624
40 Dong J. Focal loss improves the model performance on multi-label image classifications with imbalanced data. In: Proceedings of the 2nd International Conference on Industrial Control Network and System Engineering Research, 2020. 18–21
41 Zheng X, Yu Z, Chen L, et al. Multi-label contrastive focal loss for pedestrian attribute recognition. In: Proceedings of the 25th International Conference on Pattern Recognition, 2021. 7349–7356

42 Su J, Zhu M, Murtadha A, et al. ZLPR: a novel loss for multi-label classification. 2022. ArXiv:2208.02955

43 Liu B, Xu N, Lv J, et al. Revisiting pseudo-label for single-positive multi-label learning. In: Proceedings of International Conference on Machine Learning, 2023. 22249–22265

44 Cho Y, Kim D, Khan M A, et al. Mining multi-label samples from single positive labels. In: Proceedings of Advances in Neural Information Processing Systems, 2022. 15903–15916

45 Blanchard P, Higham D J, Higham N J. Accurately computing the log-sum-exp and softmax functions. IMA J Numer Anal, 2021, 41: 2311–2330

46 Fazel M, Hindi H, Boyd S P. Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In: Proceedings of the American Control Conference, 2003. 2156–2162

47 Lukasik M, Bhojanapalli S, Menon A, et al. Does label smoothing mitigate label noise? In: Proceedings of International Conference on Machine Learning, 2020. 6448–6458

48 Zhang C B, Jiang P T, Hou Q, et al. Delving deep into label smoothing. IEEE Trans Image Process, 2021, 30: 5984–5996

49 Lukov T, Zhao N, Lee G H, et al. Teaching with soft label smoothing for mitigating noisy labels in facial expressions. In: Proceedings of European Conference on Computer Vision, 2022. 648–665

50 Liu S, Niles-Weed J, Razavian N, et al. Early-learning regularization prevents memorization of noisy labels. In: Proceedings of Advances in Neural Information Processing Systems, 2020. 20331–20342

51 Kim T, Ko J, Choi J, et al. Fine samples for learning with noisy labels. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 24137–24149

52 Zheng G, Awadallah A H, Dumais S. Meta label correction for noisy label learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2021. 11053–11061

53 Wu X Z, Zhou Z H. A unified view of multi-label performance measures. In: Proceedings of International Conference on Machine Learning, 2017. 3780–3788

54 Everingham M, van Gool L, Williams C K I, et al. The Pascal visual object classes challenge 2012 (VOC2012) results. 2012, http://www.pascalnetwork.org/challenges/VOC/voc2012/workshop/index.html

55 Lin T Y, Maire M, Belongie S, et al. Microsoft COCO: common objects in context. In: Proceedings of European Conference on Computer Vision, 2014. 740–755

56 Chua T S, Tang J, Hong R, et al. NUS-WIDE: a real-world web image database from National University of Singapore. In: Proceedings of the ACM International Conference on Image and Video Retrieval, 2009. 1–9

57 Wah C, Branson S, Welinder P, et al. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report. 2011, California Institute of Technology

58 He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016. 770–778

59 Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis, 2015, 115: 211–252

60 Selvaraju R R, Cogswell M, Das A, et al. Grad-CAM: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2017. 618–626

Appendix A

For clear understanding, we provide the detailed derivation from (1) to (2).

Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_k)$, and $x_{\max} = \max(x_1, x_2, \ldots, x_k)$. Then we have

$$e^{x_{\max}} < \sum_{i=1}^{k} e^{x_i} \leqslant \sum_{i=1}^{k} e^{x_{\max}}. \tag{A1}$$

The first inequality holds by the non-negative of the exponential function, and the second inequality holds by the definition of $x_{\max}$. Taking the logarithm at each side of the above formula, then we have

$$x_{\max} < \log \sum_{i=1}^{k} e^{x_i} \leqslant \log \sum_{i=1}^{k} e^{x_{\max}} = x_{\max} + \log k. \tag{A2}$$

According to the definition of $\text{logsumexp}(\boldsymbol{x}) = \log \sum_{i=1}^{k} e^{x_i}$, we can obtain that the logsumexp is an approximation of the maximum operator, and the error is no more than $\log k$. By changing the first max operator in (1) to min and substituting the logsumexp into the second max operator, we finally obtain (2).

Besides, we also detail the derivation of $\mathcal{L}_{\text{OPML}}$ (Eq. (5) in the main text). First, we list (4) in the main text below and renumber it as

$$\max_{\theta} \left(\min_{p \in \Omega_p} s_p - \max_{n \in \Omega_n} s_n \right), \tag{A3}$$

where θ is the parameter of deep neural networks, s_p and s_n are the scores of positive and negative labels, Ω_p and Ω_n are the index sets of positive and negative labels, respectively. Note that $\min(a, b) = -\max(-a, -b)$; then we can equivalently transform Eq. (A3) into the following formulation:

$$\max_{\theta} \left(-\max_{p \in \Omega_p} (-s_p) - \max_{n \in \Omega_n} s_n \right). \tag{A4}$$

Next, by changing the first max function in (A4) to min, we can obtain

$$\min_{\theta}\left(\max_{p\in\Omega_p}(-s_p)+\max_{n\in\Omega_n}s_n\right). \tag{A5}$$

By substituting the smooth approximation of the max function, i.e., logsumexp, which is defined as $\text{logsumexp}(x_1,x_2,\ldots,x_k)=\log\sum_{i=1}^{k}e^{(x_i)}$ for any $x_i\in(-\infty,\infty)$, into (A5), we can obtain the following objective function:

$$\min_{\theta}\left(\log\left(\sum_{p\in\Omega_p}e^{(-s_p)}\right)+\log\left(\sum_{n\in\Omega_n}e^{s_n}\right)\right). \tag{A6}$$

Let $\mathcal{L}'=\log\sum_{p\in\Omega_p}e^{(-s_p)}+\log\sum_{n\in\Omega_n}e^{s_n}$ be the loss function. For the same reason as we mentioned in the main text, this loss is unbounded and the optimal tends to negative infinity, which is unstable and difficult to optimize in deep neural networks. To make this loss function bounded and stable, we also add two positive constants α and β into $\mathcal{L}'$ and achieve the final loss function, denoted by $\mathcal{L}_{\text{OPML}}$:

$$\mathcal{L}_{\text{OPML}}=\log\left(\alpha+\sum_{p\in\Omega_p}e^{(-s_p)}\right)+\log\left(\beta+\sum_{n\in\Omega_n}e^{s_n}\right). \tag{A7}$$

So far, we have provided the detailed derivation of $\mathcal{L}_{\text{OPML}}$ (Eq. (5) in the main text).