Semantic-Aware Representation Blending for Multi-Label Image Recognition with Partial Labels

Tao Pu [1], Tianshui Chen [2], Hefeng Wu [1], Liang Lin [1]*

[1] Sun Yat-Sen University, [2] Guangdong University of Technology
putao3@mail2.sysu.edu.cn, tianshuichen@gmail.com, wuhefeng@gmail.com, linliang@ieee.org

Abstract

Training the multi-label image recognition models with partial labels, in which merely some labels are known while others are unknown for each image, is a considerably challenging and practical task. To address this task, current algorithms mainly depend on pre-training classification or similarity models to generate pseudo labels for the unknown labels. However, these algorithms depend on sufficient multi-label annotations to train the models, leading to poor performance especially with low known label proportion. In this work, we propose to blend category-specific representation across different images to transfer information of known labels to complement unknown labels, which can get rid of pre-training models and thus does not depend on sufficient annotations. To this end, we design a unified semantic-aware representation blending (SARB) framework that exploits instance-level and prototype-level semantic representation to complement unknown labels by two complementary modules: 1) an instance-level representation blending (ILRB) module blends the representations of the known labels in an image to the representations of the unknown labels in another image to complement these unknown labels. 2) a prototype-level representation blending (PLRB) module learns more stable representation prototypes for each category and blends the representation of unknown labels with the prototypes of corresponding labels to complement these labels. Extensive experiments on the MS-COCO, Visual Genome, Pascal VOC 2007 datasets show that the proposed SARB framework obtains superior performance over current leading competitors on all known label proportion settings, i.e., with the mAP improvement of 4.6%, 4.6%, 2.2% on these three datasets when the known label proportion is 10%. Codes are available at https://github.com/HCPLab-SYSU/HCP-MLR-PL.

Introduction

Multi-label image recognition (MLR) (Chen et al. 2019d,b; Wu et al. 2020), which aims to find out all semantic labels from the input image, is a more challenging and practical task compared with the single-label counterpart. Due to the complexity of the input images and output label spaces, collecting a large-scale dataset with complete multi-label annotation is extremely time-consuming. To deal with this issue, recent works tend to study the task of multi-label image recognition with partial labels (MLR-PL), in which merely a few positive and negative labels are provided whereas other labels are unknown (see Figure 1). MLR-PL is more practical to real-world scenarios because it does not require complete multi-label annotations for each image.

	[a]	[b]
person	1	0
giraffe	-1	0
car	1	1
zebra	1	0
boat	-1	-1
⋮	⋮	⋮
elephant	-1	0

Figure 1: An MLR image with complete labels [a], partial labels [b], in which 1 represents the corresponding category exists, -1 represents it does not exist, and 0 represents it is unknown.

Previous works (Sun et al. 2017; Joulin et al. 2016) simply ignore the unknown labels or treat them as negative, and they adopt traditional MLR algorithms to address this task. However, it may lead to poor performance because it either loses some annotations or even incurs some incorrect labels. More recent works (Durand, Mehrasa, and Mori 2019; Huynh and Elhamifar 2020) propose to train classification or similarity models with given labels, and use these models to generate pseudo labels for the unknown labels. Despite achieving impressive progress, these algorithms depend on sufficient multi-label annotation for model training, and they suffer from obvious performance drop if decreasing the known label proportion to a small level.

Fortunately, a specific label c that is unknown in one image I^n may be known in another image I^m. We can extract the information of label c from image I^m, blend this information to image I^n, and in this way complement the unknown label c for image I^n. Previous works (Zhang et al. 2017) utilize mixup algorithm to blend two images and generate a new image with semantic information from both images to help regularize training single-label recognition models. However, a multi-label image generally has multiple semantic objects scattering over the whole image, and

*Tao Pu and Tianshui Chen contribute equally to this work and share first authorship. Corresponding author is Liang Lin.

simply blending two images lead to confusing semantic information. In this work, we design a unified semantic-aware representation blending (SARB) framework that learns and blends category-specific feature representation to complement the unknown labels. This framework does not depend on pre-trained models, and thus it can perform consistently well on all known label proportion settings.

Specifically, we first introduce a category-specific representation learning (CSRL) module (Chen et al. 2019b; Ye et al. 2020) that incorporates category semantics to guide generating category-specific representations. An instance-level representation blending (ILRB) module is designed to blend the representations of the known label c in one image I^m to the representations of the corresponding unknown label c in another image I^n. In this way, image I^n can also contain the information of label c and thus this label is complemented. This module can generate diverse blended representations to facilitate the performance but these diverse representations may also lead to unstable training. To solve this problem, a prototype-level representation blending (PLRB) module is further proposed to learn more robust representation prototypes for each category and blend the representation of unknown labels with the prototypes of the corresponding categories. In this way, we can simultaneously generate diverse and stable blended representations to complement the unknown labels and thus facilitate the MLR-PL task.

The contributions of this work are summarized into three folds: 1) We propose a semantic-aware representation blending (SARB) framework to complement unknown labels. It does not depend on pre-trained models and performs consistently well on all known label proportion settings. 2) We design the instance-level and prototype-level representation blending modules that generate diverse and stable blended feature representation to complement unknown labels. 3) We conduct extensive experiments on several large-scale MLR datasets, including Microsoft COCO (Lin et al. 2014), Visual Genome (Krishna et al. 2016) and Pascal VOC 2007 (Everingham et al. 2010), to demonstrate the effectiveness of the proposed framework. We also conduct ablative studies to analyze the actual contribution of each module for profound understanding.

Related Work

MLR with Complete/Partial Labels. Multi-label image recognition receives increasing attention in the computer vision community due to its wide application to scene recognition (Chen et al. 2019a; Zhang et al. 2020; Liu, Wu, and Lin 2015), human attribute recognition (Guo et al. 2019; Zhu et al. 2017; Chen et al. 2021b), etc. Previous works depend on object localization technology (Wei et al. 2016) or visual attention mechanism (Wang et al. 2017; Chen et al. 2018b) to discover discriminative regions and enhance feature representation to facilitate classification. Considering the guidance of semantics to visual representation learning (Chen et al. 2021a), recent works further introduce category semantics to help learn category-specific discriminative regions (Chen et al. 2019b; Wu et al. 2020), e.g., Semantic Decoupling (SD) module (Chen et al. 2019b; Wu et al. 2020), Semantic Attention Module (SAM) (Ye et al. 2020) and Class Activation Maps (CAM) (Gao and Zhou 2021). On the other hand, label correlations exist commonly among different categories and these correlations are also important for multi-label recognition. Recent works resort to graph neural networks (Abadal et al. 2022; Chen et al. 2020a) to explicitly model these correlations to learn contextualized feature representation to facilitate multi-label recognition (Chen et al. 2019d,b; Wu et al. 2020; Ye et al. 2020; Chen et al. 2020b).

Training traditional multi-label image recognition models depends on large-scale datasets with complete annotations per image. To reduce the annotation cost, the current effort (Durand, Mehrasa, and Mori 2019; Huynh and Elhamifar 2020) is dedicated to the MLR-PL task, in which merely a few labels are known while the others are known for each image. Earlier works (Sun et al. 2017; Joulin et al. 2016) formulate MLR as multiple binary classifications, and simply ignore missing labels or treat missing labels negative. Then, they train traditional multi-label models for this task, which leads to poor performance because they lose some data or even incur noisy labels. Inversely, more recent works tend to generate pseudo labels. For example, Durand et al. (Durand, Mehrasa, and Mori 2019) pre-train classification models with the given annotations and generate pseudo labels for the unknown labels based on the trained models. Then, they use both the given and updated labels to re-train the models. Huynh et al. (Huynh and Elhamifar 2020) propose to learn image-level similarity models to generate pseudo labels and progressively re-train the model similarly. However, these algorithms rely on sufficient multi-label annotations for model training, leading to poor performance when the known label proportions decrease to a low level.

Different from all these algorithms, our SARB framework learns and blends category-specific feature representation across different images to complement the unknown labels. It gets rid of pre-training models and can obtain consistently well performance on all known label settings.

Blending Regularization. Mixup (Zhang et al. 2017; Yun et al. 2019; Kim, Choo, and Song 2020) is recently proposed to blend two input images thus as to generate more diverse samples to regularize training. As a pioneer work, Zhang et al. (Zhang et al. 2017) directly perform pixel-wise blending between two images and it can obtain quite an impressive improvement for single-label image recognition. Cutmix (Yun et al. 2019) further proposes to randomly cut one region from an image and paste it to another image to generate new samples. Despite achieving impressive performance, these algorithms are very difficult to apply to the multi-label recognition scenarios, because a multi-label image inherently possesses multiple semantic objects scattering over the whole image, and simply blending two images may generate disturbed and confusing information.

Different from the mixup algorithm, the SARB framework proposes to learn and blend category-specific representation, in which the blending is performed between two representation vectors that belong to the same category. In this way, we can utilize the semantic representation of known labels to complement the representation of the unknown labels, and thus to complement these unknown labels.

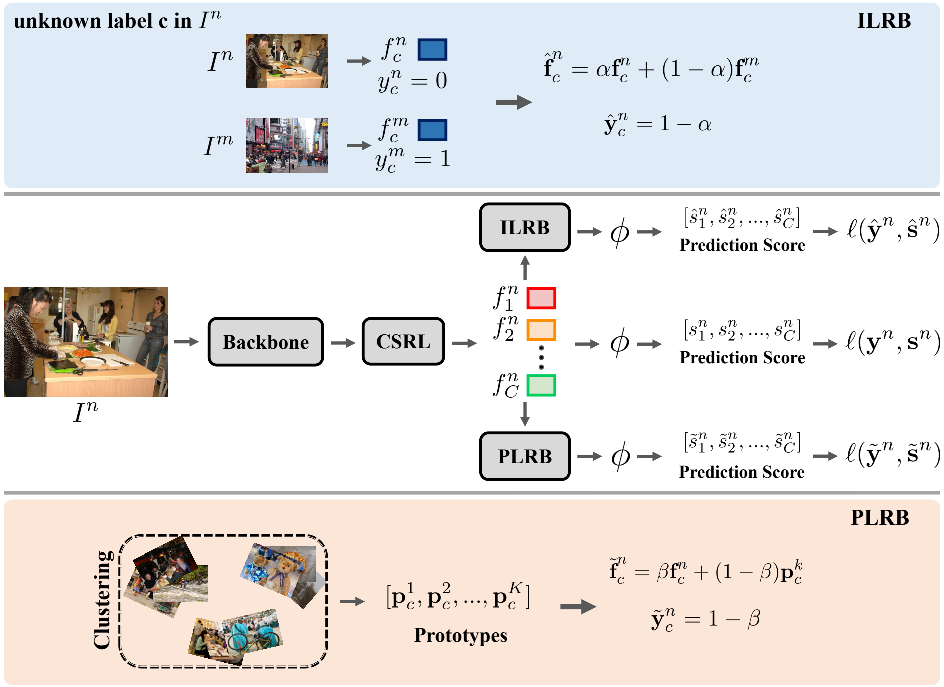

unknown label c in I^n
ILRB
I^n
f_c^n
$y_c^n = 0$
I^m
f_c^m
$y_c^m = 1$
$\hat{\mathbf{f}}_c^n = \alpha \mathbf{f}_c^n + (1-\alpha)\mathbf{f}_c^m$
$\hat{\mathbf{y}}_c^n = 1-\alpha$
ILRB
ϕ
$[\hat{s}_1^n, \hat{s}_2^n, ..., \hat{s}_C^n]$
Prediction Score
$\ell(\hat{\mathbf{y}}^n, \hat{\mathbf{s}}^n)$
I^n
Backbone
CSRL
f_1^n
f_2^n
f_C^n
ϕ
$[s_1^n, s_2^n, ..., s_C^n]$
Prediction Score
$\ell(\mathbf{y}^n, \mathbf{s}^n)$
PLRB
ϕ
$[\tilde{s}_1^n, \tilde{s}_2^n, ..., \tilde{s}_C^n]$
Prediction Score
$\ell(\tilde{\mathbf{y}}^n, \tilde{\mathbf{s}}^n)$
PLRB
Clustering
$[\mathbf{p}_c^1, \mathbf{p}_c^2, ..., \mathbf{p}_c^K]$
Prototypes
$\tilde{\mathbf{f}}_c^n = \beta \mathbf{f}_c^n + (1-\beta)\mathbf{p}_c^k$
$\tilde{\mathbf{y}}_c^n = 1-\beta$

Figure 2: An overall illustration of the proposed semantic-aware representation blending (SARB) framework. It consists of the ILRB and PLRB modules that perform instance-level and prototype-level representation blending to complement unknown labels. The classifier ϕ is shared.

Semantic-aware Representation Blending

Overview

In this section, we introduce the proposed SARB framework that consists of two complementary modules that perform instance-level and prototype-level representation blending to complement unknown labels, i.e., the ILRB and PLRB modules. The ILRB module blends the semantic representations of known labels in one image to the presentations of the unknown labels in another image to complement these unknown labels. Meanwhile, the PLRB module learns representation prototypes for each category and blends the representation of the unknown labels of the training image with the corresponding prototypes to complement these unknown labels. Finally, both the ground truth and complemented labels are used to train the multi-label models. Figure 2 illustrates an overall pipeline of the proposed framework.

Given a training image I^n, we utilize a backbone network to extract the global feature maps $\mathbf{f}^n$, and then introduce a category-specific representation learning (CSRL) module that incorporates category semantics to generate category-specific representation

$$[\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n] = \phi_{csrl}(\mathbf{f}^n), \tag{1}$$

where C is the category number. There are different algorithms to implement the CSRL module, including semantic decoupling proposed in (Chen et al. 2019b) and semantic attention mechanism proposed in (Ye et al. 2020). Then we follow previous work (Chen et al. 2019b,c, 2018a, 2021b) to use a gated neural network and a linear classifier followed by a sigmoid function to compute the probability score vectors

$$[s_1^n, s_2^n, \cdots, s_C^n] = \phi([\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n]). \tag{2}$$

Based on the learned category-specific semantic representation, the ILRB and PLRB modules are used to complement the feature representation of the unknown labels. We introduce these two modules in the following.

Instance-Level Representation Blending

Intuitively, an unknown label c in image I^n may be known in another image I^m. The ILRB module aims to blend the information of label c in image I^m to image I^n, and thus image I^n can also have the known label c. To achieve this end, we blend the representations that belong to the same category and from different images to transfer the known labels of one image to the unknown labels of the other image.

Formally, given two training images I^n and I^m, whose learned semantic representation vectors are $[\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n]$ and $[\mathbf{f}_1^m, \mathbf{f}_2^m, \cdots, \mathbf{f}_C^m]$, and label vectors are $y^n = \{y_1^n, y_2^n, \cdots, y_C^n\}$ and $y^m = \{y_1^m, y_2^m, \cdots, y_C^m\}$, we blend the semantic representations and labels for each category. For category c, the blending process can be formulated as

$$\hat{\mathbf{f}}_c^n = \begin{cases} \alpha \mathbf{f}_c^n + (1-\alpha)\mathbf{f}_c^m & y_c^n = 0, y_c^m = 1, \\ \mathbf{f}_c^n & otherwise, \end{cases} \tag{3}$$

$$\hat{\mathbf{y}}_c^n = \begin{cases} 1-\alpha & y_c^n = 0, y_c^m = 1, \\ y_c^n & otherwise, \end{cases} \tag{4}$$

where α is the learnable parameter and its initial value is set to 0.5. We repeat the above blending process for all categories, and reformulate them as matrix operations for efficient computing

$$\hat{\mathbf{F}}^n = A\mathbf{F}^n + (1 - A)\mathbf{F}^m, \tag{5}$$

$$\hat{\mathbf{y}}^n = A\mathbf{y}^n + (1 - A)\mathbf{y}^m, \tag{6}$$

where $A = [\alpha_1, \alpha_2, \cdots, \alpha_C]$ is a parameter vector; $\mathbf{F}^n = [\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n]$ and $\mathbf{F}^m = [\mathbf{f}_1^m, \mathbf{f}_2^m, \cdots, \mathbf{f}_C^m]$ are the feature matrices for all categories of image n and m; $\hat{\mathbf{F}}^n = [\hat{\mathbf{f}}_1^n, \hat{\mathbf{f}}_2^n, \cdots, \hat{\mathbf{f}}_C^n]$ and $\hat{\mathbf{y}}^n = [\hat{\mathbf{y}}_1^n, \hat{\mathbf{y}}_2^n, \cdots, \hat{\mathbf{y}}_C^n]$ are the blended semantic representation and label matrix. Then, we use a gated graph neural network and linear classifier followed by sigmoid function to compute the probability score vector $\hat{\mathbf{s}}^n$.

Prototype-Level Representation Blending

Although the ILRB module can obviously improve the performance, it may disturb the training process because it generates many diverse blended representation for training, especially when the known label proportion is low. To deal with this issue, we further design a PLRB module that learns to generate more stable representation prototypes for each category and blend the representation of unknown labels in image I^n with the prototypes of corresponding categories.

The prototypes are used to describe the overall representation of the corresponding category. For each category c, we first select all the images that have the known label c, and then extract the representations of this category, resulting in the feature vectors $[\mathbf{f}_c^1, \mathbf{f}_c^2, \cdots, \mathbf{f}_c^{N_c}]$. Then, we simply use the K-means algorithm to cluster these feature vectors into K prototypes, i.e., $P_c = [\mathbf{p}_c^1, \mathbf{p}_c^2, ..., \mathbf{p}_c^K]$.

It is expected that the representations of the same category is similar, and thus it can learn more compact distribution to better compute the prototypes for each category. To achieve this end, we utilize contrastive loss for increasing the similarity between $\mathbf{f}_c^n$ and $\mathbf{f}_c^m$ if images n and m have the same existing category c, and decreasing the similarity otherwise. Thus, it can be formulated as

$$\ell_c^{n,m} = \begin{cases} 1 - cosine(\mathbf{f}_c^n, \mathbf{f}_c^m) & y_c^n = 1, y_c^m = 1, \\ 1 + cosine(\mathbf{f}_c^n, \mathbf{f}_c^m) & otherwise, \end{cases} \tag{7}$$

where $cosine(\cdot, \cdot)$ represents a function that computes the cosine similarity between the input. The final contrastive loss can be formulated as

$$\mathcal{L}_{cst} = \sum_{n=1}^{N} \sum_{m=1}^{N} \sum_{c=1}^{C} \ell_c^{n,m}. \tag{8}$$

Given an input image I^n whose learned semantic representation vectors $[\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n]$ and corresponding label vectors $y^n = \{y_1^n, y_2^n, \cdots, y_C^n\}$, we randomly select a label c that is unknown, then randomly select a prototype from P_c and blend it with the representation of label c, formulated as

$$\tilde{\mathbf{f}}_c^n = \begin{cases} \beta \mathbf{f}_c^n + (1 - \beta)\mathbf{p}_c^k & c = random(\{c | y_c^n = 0\}) \\ \mathbf{f}_c^n & otherwise, \end{cases} \tag{9}$$

$$\tilde{\mathbf{y}}_c^n = \begin{cases} 1 - \beta & c = random(\{c | y_c^n = 0\}) \\ y_c^n & otherwise, \end{cases} \tag{10}$$

where β is a also learnable parameter, and it is initialized as 0.5; $random(\dot{})$ represents a random sampling function which means we randomly choose one unknown category to blend semantic representation per image; k is randomly sampled in $[1, ..., K]$ and obeys uniform distribution. We repeat the above blending process for all categories, and reformulate them as matrix operations for efficient computing:

$$\tilde{\mathbf{F}}^n = B\mathbf{F}^n + (1 - B)\mathbf{P}^k, \tag{11}$$

$$\tilde{\mathbf{y}}^n = B\mathbf{y}^n + (1 - B), \tag{12}$$

where $B = [\beta_1, \beta_2, \cdots, \beta_C]$ is a parameter vector; $\mathbf{F}^n = [\mathbf{f}_1^n, \mathbf{f}_2^n, \cdots, \mathbf{f}_C^n]$ and $\mathbf{P}^k = [\mathbf{p}_1^k, \mathbf{p}_2^k, \cdots, \mathbf{p}_C^k]$ are the feature matrices for all categories of image n and prototype k; $\tilde{\mathbf{F}}^n = [\tilde{\mathbf{f}}_1^n, \tilde{\mathbf{f}}_2^n, \cdots, \tilde{\mathbf{f}}_C^n]$ and $\tilde{\mathbf{y}}^n = [\tilde{\mathbf{y}}_1^n, \tilde{\mathbf{y}}_2^n, \cdots, \tilde{\mathbf{y}}_C^n]$ are the blended semantic representation and label matrix. Then, we use a gated graph neural network and linear classifier followed by the sigmoid function to compute the probability score vector $\tilde{\mathbf{s}}^n$.

Optimization

Following previous works, we utilize the partial binary cross entropy loss as the objective function for supervising the network. In particular, given the predicted probability score vector $\mathbf{s}^n = \{s_1^n, s_2^n, \cdots s_C^n\}$ and the ground truth of known labels, the objective function can be defined as

$$\begin{aligned} \ell(\mathbf{y}^n, \mathbf{s}^n) = & \frac{1}{\sum_{c=1}^{C} |y_c^n|} \sum_{c=1}^{C} [\mathbf{1}(y_c^n = 1) \log(s_c^n) \\ & + \mathbf{1}(y_c^n = -1) \log(1 - s_c^n)], \end{aligned} \tag{13}$$

where $\mathbf{1}[\cdot]$ is an indicator function whose value is 1 if the argument is positive and is 0 otherwise.

Similarly, we adopt the partial binary cross entropy loss as the objective function for supervising the ILRB module and PLRB module, i.e., $\ell(\hat{\mathbf{y}}^n, \hat{\mathbf{s}}^n)$ and $\ell(\tilde{\mathbf{y}}^n, \tilde{\mathbf{s}}^n)$. Therefore, the final classification loss is defined as summing the three losses over all samples, formulated as

$$\mathcal{L}_{cls} = \sum_{n=1}^{N} [\ell(\mathbf{y}^n, \mathbf{s}^n) + \ell(\hat{\mathbf{y}}^n, \hat{\mathbf{s}}^n) + \ell(\tilde{\mathbf{y}}^n, \tilde{\mathbf{s}}^n)]. \tag{14}$$

Finally, we sum over the classification and contrastive losses of all samples to obtain the final loss, formulated as

$$\mathcal{L} = \mathcal{L}_{cls} + \lambda \mathcal{L}_{cst}. \tag{15}$$

Here, λ is a balance parameter that ensures the contrastive loss $\mathcal{L}_{cst}$ has a comparable magnitude with the classification loss $\mathcal{L}_{cls}$. Since $\mathcal{L}_{cst}$ is much larger than $\mathcal{L}_{cls}$, we set λ to 0.05 in the experiments.

Experiments

Experimental Setting

Implementation Details For fair comparison, we follow previous work to adopt the ResNet-101 (He et al. 2016) as the backbone to extract global feature maps. We initialize its parameters with those pre-trained on the ImageNet (Deng et al. 2009) dataset while initializing the parameters of all newly-added layers randomly. We fix the parameters of the previous 91 layers of ResNet-101, and train the other layers in an end-to-end manner. During training, we use the Adam algorithm (Kingma and Ba 2015) with a batch size of 16, momentums of 0.999 and 0.9, and a weight decay of 5×10^{-4}. We set the initial learning rate as 10^{-5} and divide it by 10 after every 10 epochs. It is trained with 20 epochs in total. For data augmentation, the input image is resized to 512×512, and we randomly choose a number from {512, 448, 384, 320, 256} as the width and height to crop patch. Finally, the cropped patch is further resized to 448×448. Besides, random horizontal flipping is also used. To stabilize the training process, we start to use the ILRB and PLRB modules at epoch 5, and re-compute prototypes of each category for every 5 epochs. During inference, the ILRB and PLRB modules are removed, and the image is resized to 448×448 for evaluation.

Dataset We conduct experiments on the MS-COCO (Lin et al. 2014), Visual Genome (Krishna et al. 2016), and Pascal VOC 2007 (Everingham et al. 2010) datasets for fair comparison. MS-COCO covers 80 daily-lift categories, which contains 82,801 images as the training set and 40,504 images as the validation set. Pascal VOC 2007 contains 9,963 images from 20 object categories, and we follow previous works to use the trainval set for training and the test set for evaluation. Visual Genome contains 108,249 images from 80,138 categories, and most categories have very few samples. In this work, we select the 200 most frequent categories to obtain a VG-200 subset. Moreover, since there is no train/val split, we randomly select 10,000 images as the test set and the rest 98,249 images are used as the training set. The train/test set will be released for further research.

Since all the datasets have complete labels, we follow the setting of previous works (Durand, Mehrasa, and Mori 2019; Huynh and Elhamifar 2020) to randomly drop a certain proportion of positive and negative labels to create partially annotated datasets. In this work, the proportions of dropped labels vary from 90% to 10%, resulting in known labels proportion of 10% to 90% .

Evaluation Metric For a fair comparison, we adopt the mean average precision (mAP) over all categories for evaluation under different proportions of known labels. And we also compute average mAP over all proportions for a more comprehensive evaluation. Moreover, we follow most previous MLR works (Chen et al. 2019b) to adopt the overall and per-class precision, recall, F1-measure (i.e., OP, OR, OF1, CP, CR, and CF1) for more comprehensive evaluation. We present the formulas of these metrics and detailed results in the supplementary material due to the paper limit.

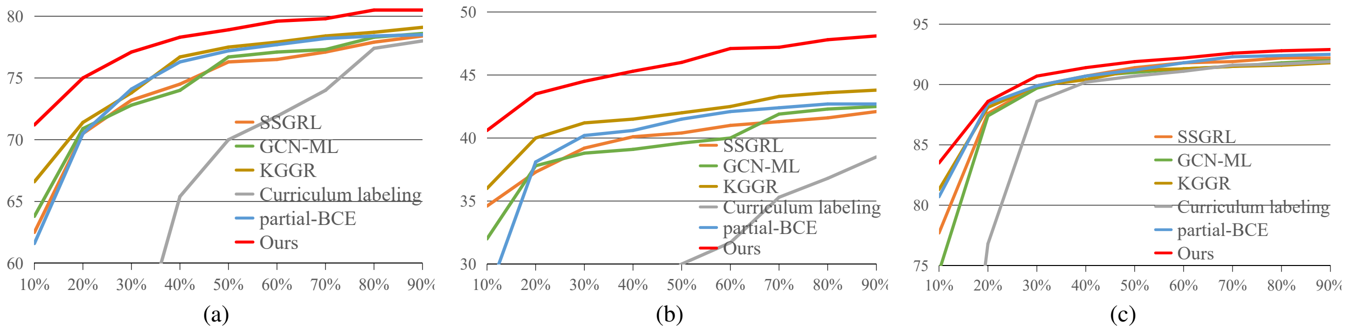
Comparison with the State-of-the-art Algorithms

To evaluate the effectiveness of the proposed SARB framework, we compare it with both the conventional MLR and current MLR-PL algorithms:
1) Conventional MLR Algorithms: semantic-specific graph representation learning (SSGRL) (Chen et al. 2019b), multi-label image recognition graph convolution network (GCN-ML) (Chen et al. 2019d), knowledge-guided graph routing (KGGR) (Chen et al. 2020b). Through exploring label dependencies or capturing semantic information, these methods achieve state-of-the-art performance on the traditional MLR task. For fair comparisons, we adapt these methods to address the MLR-PL task by replacing BCE loss with partial BCE loss.
2) Current MLR-PL Algorithms: partial binary cross entropy loss (partial-BCE) (Durand, Mehrasa, and Mori 2019), Curriculum Labeling (Durand, Mehrasa, and Mori 2019). It is worth noting that partial-BCE not only is easy to implement but also achieves state-of-the-art performance on the MLR-PL task.

Performance on MS-COCO We first present the performance comparisons on MS-COCO in Table 1 and Figure 3(a). Our SARB framework obtains the overall best performance over current state-of-the-art algorithms. As shown in Table 1, it achieves the average mAP, OF1, and CF1 of 77.9%, 76.5%, and 72.2%, outperforming the previous best-performing KGGR algorithm by 2.3%, 2.8%, and 2.5%, respectively. As shown in Figure 3(a), the SARB framework also achieves better mAP over all known label proportion settings. It is noteworthy that the SARB framework obtains more obvious performance improvement when decreasing the known label proportions. For example, the mAP improvements over the previous best KGGR algorithm are 1.4% and 4.6% when using 90% and 10% known labels, respectively. These comparisons demonstrate that the SARB framework can be adapted to different proportion settings as it does not depend on pre-trained models.

Performance on VG-200 As previously discussed, VG-200 is a more challenging benchmark that covers much more categories. Thus, current works achieve quite poor performance. As shown in Table 1, the previous best-performing KGGR algorithm obtains the average mAP, OF1, and CF1 of 41.5%, 41.2%,and 33.6%. In this scenario, our SARB framework exhibits much more obvious performance improvement. Its average mAP, OF1, and CF1 are 45.6%, 45.0%, and 37.4%, outperforming the KGGR algorithm by 4.1%, 3.8%, and 3.8%. We also present the mAP comparisons over different known proportion settings in Figure 3(b). Compared with current algorithms, we find that our framework achieves the mAP improvement of more than 3.3% on all known label proportion settings.

Performance on Pascal VOC 2007 Pascal VOC 2007 is the most widely used dataset for evaluating multi-label image recognition. Here, we also present the performance comparisons on this dataset in Table 1 and Figure 3(c). As this dataset covers merely 20 categories, it is a much simpler dataset and current algorithms can also achieve quite well performance. However, our SARB framework can still achieve consistent improvement. As shown, it improves the average mAP, OF1, and CF1 by 0.7%, 0.5%, and 1.1%. In addition, it exhibits a similar phenomenon that the mAP improvement is more obvious when using the fewer known labels, with 0.4% and 2.8% mAP improvement using 90% and 10% known label proportions as shown in Figure 3(c).



(a) (b) (c)

Figure 3: The mAP of our SARB framework and current state-of-the-art competitors on the settings of known label proportions of 10% to 90% on the MS-COCO (left), VG-200 (middle) and Pascal VOC 2007 (right) datasets. Best viewed in color.

Datasets	Methods	Avg. mAP	Avg. OP	Avg. OR	Avg. OF1	Avg. CP	Avg. CR	Avg. CF1
MS-COCO	SSGRL	74.1	86.3	64.8	73.9	82.1	58.4	68.1
	GCN-ML	74.4	85.2	64.2	73.1	81.8	58.9	68.4
	KGGR	75.6	84.0	65.6	73.7	81.4	60.9	69.7
	Curriculum labeling	60.7	87.8	51.0	61.9	60.9	40.4	48.3
	partial-BCE	74.7	**86.7**	64.7	74.0	**83.1**	58.9	68.8
	Ours	**77.9**	86.6	**68.6**	**76.5**	82.9	**64.1**	**72.2**
VG-200	SSGRL	39.7	69.9	25.9	37.8	45.3	18.3	26.1
	GCN-ML	39.3	64.1	28.2	38.7	44.6	18.2	25.6
	KGGR	41.5	64.5	30.5	41.2	54.8	25.8	33.6
	Curriculum labeling	28.4	66.4	15.4	23.6	20.4	7.6	10.9
	partial-BCE	39.8	69.7	24.6	36.1	44.3	18.1	25.7
	Ours	**45.6**	**70.1**	**33.2**	**45.0**	**56.8**	**27.8**	**37.4**
Pascal VOC 2007	SSGRL	89.5	91.2	**84.4**	87.7	87.8	**81.4**	84.5
	GCN-ML	88.9	92.2	83.0	87.3	89.7	80.1	84.6
	KGGR	89.7	90.5	82.9	86.5	88.5	81.4	84.7
	Curriculum labeling	84.1	92.7	78.2	83.8	79.5	71.7	75.4
	partial-BCE	90.0	91.8	84.3	87.9	88.8	81.3	84.8
	Ours	**90.7**	**93.0**	83.6	**88.4**	**90.4**	81.1	**85.9**

Table 1: Average mAP, OP, OR, OF1 and CP, CR, CF1 of the proposed SARB framework and current state-of-the-art competitors for multi-label recognition with partial labels on the MS-COCO, VG-200 and Pascal VOC 2007 datasets. The best results are highlighted in bold.

Ablative Studies

In this section, we conduct ablative studies to analyze the actual contributions of each module in our SARB framework.

Analysis of the CSRL Module

The CSRL module is used to extract category-specific feature representation and is a basic module of the proposed framework. There are different kinds of algorithms to implement the CSRL module, in which semantic decoupling (SD) (Chen et al. 2019b) and semantic attention mechanism (SAM) (Ye et al. 2020) are two choices that obtain state-of-the-art performance for the traditional MLR task. Here, we conduct an experiment to compare these two algorithms and present the results in Table 2. It shows that using the two algorithms obtain comparable performance. More concretely, using SD obtains slightly better performance than using SAM, with an average mAP improvement of 0.3%, 0.2%, and 0.1% on the three datasets. Thus, we use the SD to implement the CSRL module for all other experiments.

Current mixup (Zhang et al. 2017) simply performs position-wise blending to generate new samples to regularize training. In this part, we further conduct two baseline algorithms that perform position-wise blending in image space and feature space (namely IP-Mixup and FM-Mixup) to verify the benefit of learning category-specific feature representation. As shown in Table 2, both two baseline algorithms achieve comparable performance with the SSGRL baselines as such simple blending can not provide additional information. Compared with the SARB using CSRL, IP-Mixup suffers from the average mAP degration of 3.6%, 5.9%, and 1.0%, while FM-Mixup suffers from the average mAP degration of 3.8%, 6.0%, and 1.1% on the three datasets, respectively.

Contribution of the SARB Module

As we use the SD algorithm to implement the CSRL module and gated neural network for classification, SSGRL

Datasets Methods	MS-COCO	VG-200	VOC2007
Ours w/ SAM	77.6	45.4	90.6
Ours w/ SD	77.9	45.6	90.7
IP-Mixup	74.3	39.7	89.7
FM-Mixup	74.1	39.6	89.6
SSGRL	74.1	39.7	89.5
Ours ILRB	77.3	44.9	90.2
Ours ILRB fixed α	76.9	44.5	89.8
Ours PLRB	77.3	44.9	90.4
Ours PLRB fixed β	76.9	44.6	90.2
Ours	77.9	45.6	90.7

Table 2: Comparison of average mAP of our framework with SAM module (Ours w/ SAM), our framework with SD module (Ours w/ SD), SSGRL with mixup on image pixel level (IP-Mixup), SSGRL with mixup on feature map level (FM-Mixup), the baseline SSGRL, our framework merely using ILRB module (Our ILRB), our framework merely using ILRB module with fixed α (Ours ILRB fixed α), our framework merely using PLRB module (Ours PLRB), our framework merely using PLRB module with fixed β (Ours PLRB fixed β) and our framework (Ours) on the MS-COCO, VG-200 and Pascal VOC 2007 datasets.

(Chen et al. 2019b) is the baseline of the proposed framework. Here, we emphasize the comparisons with SSGRL to demonstrate the effectiveness of SARB. As shown in Table 2, SSGRL obtains the average mAPs of 74.1%, 39.7%, and 89.5% on the MS-COCO, VG-200, and Pascal VOC datasets. By integrating the SARB module, it boosts the average mAP to 77.9%, 45.6%, and 90.7% on the three datasets, with the mAP improvement of 3.8%, 5.9%, and 1.2%, respectively.

SARB consists of the instance-level and prototype-level representation blending modules. In the following, we further conduct experiments to analyze these two modules for more in-depth understanding.

Analysis of the ILRB Module

To analyze the actual contribution of the ILRB module, we conduct experiments that merely use this module (namely, Ours ILRB) and compare it with the SSGRL baseline on the MS-COCO, VG-200, Pascal VOC 2007 datasets. As shown in Table 2, it obtains an average mAP of 77.3%, 44.9%, 90.2% on MS-COCO, VG-200, Pascal VOC 2007, with the mAP improvement of 3.2%, 5.2%, and 0.7%, respectively.

ILRB contains an crucial parameter α that controls the ratio of instance-level mix-up. However, it is impractical and exhausting to find a best value for different datasets and different settings. In this work, we set α as a learnable parameter to adaptively learn the best value via standard backpropagation. To verify its contribution, we conduct an experiment to compare with the baseline using a fixed α of 0.5. As shown in Table 2, using a fixed value of 0.5 decreases the average mAPs from 77.3%, 44.9%, and 90.2% to 76.9%, 44.5%, and 89.8%, respectively.

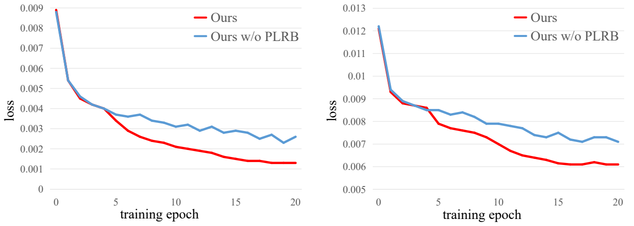


Figure 4: Analysis of the effect on PLRB. These experiments are conducted on MS-COCO (left) and VG-200 (right).

Analysis of the PLRB Module

Similarly, PLRB is another module that plays a key role, and in this part, we also analyze its effectiveness by comparing the performance with and without it. As shown in Table 2. Adding the PLRB module to the baseline SSGRL leads to 3.2%, 5.2%, and 0.9% mAP improvement on the MS-COCO, VG-200, and Pascal VOC 2007 datasets. As previously suggested, the PLRB module can help to generate stable blended representations to complement unknown labels, which leads to more stable training. To validate this point, we further visualize the loss of the training process in Figure 4. It can be observed that the loss is choppy without the PLRB module, and adding this module can stabilize the training process.

The parameter β is a learnable parameter that is adaptively learned for different datasets and settings. Here, we also conduct experiments to compare with the setting that fixes β to 0.5 on the MS-COCO, VG-200, Pascal VOC 2007 datasets. As presented in Table 2, it obtains an average mAP of 76.9%, 44.6%, 90.2% on these three datasets, with the slight degeneration of 0.4%, 0.3% and 0.2%.

Conclusion

In this work, we present a new perspective to complement the unknown labels by blending category-specific feature representation to address the MLR-PL task. It does not depend on sufficient annotations and thus can obtain superior performance on all known label proportion settings. Specifically, it consists of an ILRB module that blends instance-level representation of known labels to complement the representation of corresponding unknown labels and a PLRB module that leans and blends prototype-level representations to complement the representation of corresponding unknown labels. It can simultaneously generate diverse and stable blended representations to complement the unknown labels and thus facilitate the MLR-PL task. Extensive experiments on the MS-COCO, VG-200, and Pascal VOC demonstrate its superiority over current algorithms.

Acknowledgements

This work was supported by National Natural Science Foundation of China (No. 61876045, 61836012 and 62002069), the Natural Science Foundation of Guangdong Province (No. 2017A030312006) and Guangdong Provincial Basic Research Program (No. 102020369).

References

Abadal, S.; Jain, A.; Guirado, R.; López-Alonso, J.; and Alarcón, E. 2022. Computing Graph Neural Networks: A Survey from Algorithms to Accelerators. *ACM Computing Surveys*, 54(9): 191:1–191:38.

Chen, L.; Zhan, W.; Tian, W.; He, Y.; and Zou, Q. 2019a. Deep integration: A multi-label architecture for road scene recognition. *IEEE Transactions on Image Processing*, 28(10): 4883–4898.

Chen, R.; Chen, T.; Hui, X.; Wu, H.; Li, G.; and Lin, L. 2020a. Knowledge Graph Transfer Network for Few-Shot Recognition. In *Proceedings of Thirty-Fourth AAAI Conference on Artificial Intelligence*, 10575–10582.

Chen, S.; Xie, G.; Liu, Y.; Peng, Q.; Sun, B.; Li, H.; You, X.; and Shao, L. 2021a. HSVA: Hierarchical Semantic-Visual Adaptation for Zero-Shot Learning. In *Proceedings of Thirty-Fifth Conference on Neural Information Processing Systems (NeurIPS)*.

Chen, T.; Lin, L.; Chen, R.; Wu, Y.; and Luo, X. 2018a. Knowledge-Embedded Representation Learning for Fine-Grained Image Recognition. In *IJCAI*, 627–634.

Chen, T.; Lin, L.; Hui, X.; Chen, R.; and Wu, H. 2020b. Knowledge-Guided Multi-Label Few-Shot Learning for General Image Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, T.; Pu, T.; Wu, H.; Xie, Y.; Liu, L.; and Lin, L. 2021b. Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Chen, T.; Wang, Z.; Li, G.; and Lin, L. 2018b. Recurrent Attentional Reinforcement Learning for Multi-label Image Recognition. In *Proceedings of AAAI Conference on Artificial Intelligence*, 6730–6737.

Chen, T.; Xu, M.; Hui, X.; Wu, H.; and Lin, L. 2019b. Learning semantic-specific graph representation for multi-label image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 522–531.

Chen, T.; Yu, W.; Chen, R.; and Lin, L. 2019c. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 6163–6171.

Chen, Z.-M.; Wei, X.-S.; Wang, P.; and Guo, Y. 2019d. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 5177–5186.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 248–255.

Durand, T.; Mehrasa, N.; and Mori, G. 2019. Learning a deep convnet for multi-label classification with partial labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 647–657.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2): 303–338.

Gao, B.-B.; and Zhou, H.-Y. 2021. Learning to Discover Multi-Class Attentional Regions for Multi-Label Image Recognition. *IEEE Transactions on Image Processing*.

Guo, H.; Zheng, K.; Fan, X.; Yu, H.; and Wang, S. 2019. Visual attention consistency under image transforms for multi-label image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 729–739.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Huynh, D.; and Elhamifar, E. 2020. Interactive multi-label CNN learning with partial labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 9423–9432.

Joulin, A.; Van Der Maaten, L.; Jabri, A.; and Vasilache, N. 2016. Learning visual features from large weakly supervised data. In *ECCV*, 67–84.

Kim, J.-H.; Choo, W.; and Song, H. O. 2020. Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In *Proceedings of International Conference on Machine Learning (ICML)*, 5275–5285.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of 3rd International Conference on Learning Representations (ICLR), San Diego, USA, May 7-9, 2015*.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; Bernstein, M.; and Fei-Fei, L. 2016. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *arXiv preprint arXiv:1602.07332*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*, 740–755.

Liu, N.; Wu, H.; and Lin, L. 2015. Hierarchical Ensemble of Background Models for PTZ-Based Video Surveillance. *IEEE Transactions on Cybernetics*, 45(1): 89–102.

Sun, C.; Shrivastava, A.; Singh, S.; and Gupta, A. 2017. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 843–852.

Wang, Z.; Chen, T.; Li, G.; Xu, R.; and Lin, L. 2017. Multi-label Image Recognition by Recurrently Discovering Attentional Regions. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 464–472.

Wei, Y.; Xia, W.; Lin, M.; Huang, J.; Ni, B.; Dong, J.; Zhao, Y.; and Yan, S. 2016. HCP: A flexible CNN framework for multi-label image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(9): 1901–1907.

Wu, X.; Chen, Q.; Li, W.; Xiao, Y.; and Hu, B. 2020. AdaHGNN: Adaptive Hypergraph Neural Networks for Multi-Label Image Classification. In *Proceedings of the 28th ACM International Conference on Multimedia (ACMMM)*, 284–293.

Ye, J.; He, J.; Peng, X.; Wu, W.; and Qiao, Y. 2020. Attention-driven dynamic graph convolutional network for multi-label image recognition. In *ECCV*, 649–665.

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 6023–6032.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhang, W.; Wang, X. E.; Tang, S.; Shi, H.; Shi, H.; Xiao, J.; Zhuang, Y.; and Wang, W. Y. 2020. Relational graph learning for grounded video description generation. In *Proceedings of ACM International Conference on Multimedia (ACMMM)*, 3807–3828.

Zhu, J.; Liao, S.; Lei, Z.; and Li, S. Z. 2017. Multi-label convolutional neural network based pedestrian attribute classification. *Image and Vision Computing*, 58: 224–229.