Boosting Multi-Label Image Classification with Complementary Parallel Self-Distillation

Jiazhi Xu[1], **Sheng Huang**[1*], **Fengtao Zhou**[1], **Luwen Huangfu**[2], **Daniel Zeng**[3] and **Bo Liu**[4]
[1]School of Big Data and Software Engineering, Chongqing University
[2]Fowler College of Business & CHDMA, San Diego State University,
[3]Institute of Automation, Chinese Academy of Sciences,
[4]JD Finance America Corporation
{xujiazhi, huangsheng, zft}@cqu.edu.cn, lhuangfu@sdsu.edu, dajun.zeng@ia.ac.cn, kfliubo@gmail.com

Abstract

Multi-Label Image Classification (MLIC) approaches usually exploit label correlations to achieve good performance. However, emphasizing correlation like co-occurrence may overlook discriminative features and lead to model overfitting. In this study, we propose a generic framework named Parallel Self-Distillation (PSD) for boosting MLIC models. PSD decomposes the original MLIC task into several simpler MLIC sub-tasks via two elaborated complementary task decomposition strategies named Co-occurrence Graph Partition (CGP) and Dis-occurrence Graph Partition (DGP). Then, the MLIC models of fewer categories are trained with these sub-tasks in parallel for respectively learning the joint patterns and the category-specific patterns of labels. Finally, knowledge distillation is leveraged to learn a compact global ensemble of full categories with these learned patterns for reconciling the label correlation exploitation and model overfitting. Extensive results on MS-COCO and NUS-WIDE datasets demonstrate that our framework can be easily plugged into many MLIC approaches and improve performances of recent state-of-the-art approaches. The source code is released at https://github.com/Robbie-Xu/CPSD.

1 Introduction

Natural images often contain multiple visual objects, which can be characterized by a set of image labels. Multi-label image classification (MLIC) task is to recognize all these objects, which is highly relevant to other vision tasks such as object detection, image retrieval, and semantic segmentation.

Most existing MLIC research works focus on exploiting the label correlation property, which distinguishes it from the single-label image classification problem. Label correlation exploitation strategies, such as pair-wise and high-order label correlation have been extensively studied. Deep learning-based approaches, such as RNN [Wang *et al.*, 2016], graph model [Chen *et al.*, 2019a; Chen *et al.*, 2019c; Nguyen *et al.*, 2021] and attention mechanism [Gao and Zhou, 2021] are widely employed to encode the image label correlation, yielding decent performance.

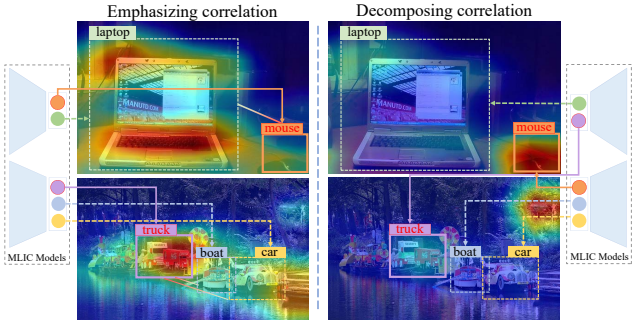

Figure 1: **Class Activation Map (CAM) of class *mouse* and *truck***. Both target labels are marked in red. Overemphasizing the correlation by specifically putting the *mouse* and *laptop* categories together to train an MLIC model leads to model overfitting because the MLIC model infers the label *mouse* only based on the features of a *laptop*, e.g. upper left. On the contrary, if we train individual models parallelly for those two labels, the model will capture the category-specific features, e.g., upper right. However, when the object is not well exposed, the model needs to rely on the label-dependence knowledge to mine the category-specific feature. Training together succeeds in recognizing label *truck* with the help of co-occurring *car* and *boat*, e.g., lower left. It is quite challenging to reconcile the model overfitting and the label correlation exploitation.

Although label correlation is a useful feature for the MLIC problem, it can also be misleading due to model overfitting. As illustrated by Figure 1, emphasizing co-occurrence leads to an inference of targets primarily from co-occurring objects, which may not represent all cases. On the contrary, decomposing this correlation leads to learning discriminative features of the target itself, which may also fail in classification due to the lack of context. Therefore, in addition to capturing label co-occurrence, our proposed framework also includes the discriminative features of individual labels, represented by dis-occurrence. Furthermore, because of its multi-class nature, the problem complexity of an MLIC task, in terms of the prediction space grows exponentially as the category number increases, e.g., the prediction result has 2^c possibilities for a c-class MLIC problem. When the image category

*Corresponding Author

number is higher, the model learning task becomes more challenging and issues with model overfitting are more likely to occur. One prominent algorithm branch to reduce the complexity is to decompose the original problem into a set of binary problems with common strategies like one-vs-one and one-vs-all [Galar *et al.*, 2011], or advanced strategies like D-Chooser [Chen *et al.*, 2021].

In this paper, we adopt this divide-and-conquer thought and propose a generic MLIC framework that addresses both model overfitting and label dependency modelling. We first decompose an MLIC task into several simpler sub-tasks, each with fewer object categories. Individual models are trained in parallel for tackling these sub-tasks. After that, knowledge distillation, an effective way to learn a compact model with generalization capability from model ensemble [Gou *et al.*, 2021], is conducted to learn a global model containing all object categories. The label decomposition reduces the complexity of each sub-task, which helps each individual model learn more representative features. In the model distillation process, those sub-models serve as teachers whose logit outputs are utilized as the soft targets to supervise the learning of a model which contains all categories with the same architecture as those teachers (i.e., self-distillation). These soft targets function as a label smoothing regularizer [Yuan *et al.*, 2020] for a better optimization. The contributions of our method are summarized as follows:

- A generic MLIC framework, called Parallel Self-Distillation (PSD) is proposed. With proper task-decomposition strategy, the original MLIC model training task is formulated as a multiple sub-model parallelly training task, then these sub-models are distilled into a global model.We demonstrate that our framework can be flexibly applied to existing MLIC models and improve their performances.
- Two strategies, namely Co-occurrence Graph Partition (CGP) and Dis-occurrence Graph Partition (DGP), are elaborated, for decomposing the MLIC task via label partition. They model the label correlation by two complementary graphs. The co-occurrence graph models the label correlation, based on which the spectral clustering result tends to assign co-occurring labels into the same cluster. This induces the individually trained sub-model to learn the joint pattern of these co-occurring classes. While the dis-occurrence graph assigns labels without co-occurrence into one task for learning the category-specific patterns. These two complementary strategies are simultaneously leveraged in PSD to reconcile the model overfitting and label correlation exploitation.

We conduct extensive experiments on two widely-used MLIC datasets, MS-COCO and NUS-WIDE. Experimental results demonstrate that our framework can be plugged into different approaches to boost the performance without increasing complexity. We also visualize the implicit attention of our framework to expose the overfitting of co-occurrence and demonstrate the effectiveness of our approach.

2 Related Works

Multi-label image classification (MLIC) is different from single-label image classification in that it relies on the label correlation property. Many MLIC approaches have been devoted to exploiting this property. For example, the methods proposed in [Chen *et al.*, 2019a; Chen *et al.*, 2019c] build label correlation graphs and adopt GNN for label feature learning. Sample imbalance is another issue of MLIC. In [Wu *et al.*, 2020], a distribution-balanced loss is proposed. It re-balances the training sample weights and designs a negative-tolerant regularization which can avoid over-suppression caused by the dominance of negative classes. An asymmetric loss is proposed in [Ridnik *et al.*, 2021], where the contribution of positive samples is maintained. Vision Transformer [Dosovitskiy *et al.*, 2020] has recently been introduced to MLIC not only because of its strong feature extraction capability, but also because the self-attention mechanism can capture rich patterns between visual features and class label tokens [Lanchantin *et al.*, 2021; Liu *et al.*, 2021]. In this work, we test the performance of our proposed framework within Transformer with a naive Transformer encoder.

Knowledge distillation was initially proposed to transfer knowledge from large complex networks to slimmer networks in order to retain the performance of the large network with less computation and model size [Hinton *et al.*, 2015]. [Zhang *et al.*, 2019] find that distilling a pretrained model with the same architecture can boost the model performance. The technique is called self-distillation. Zhou *et al.* investigate the bias-variance tradeoff brought by distillation and propose to use weighted soft labels that enable a sample-wise bias-variance tradeoff [Zhou *et al.*, 2021c]. In [Xiang *et al.*, 2020], a multiple experts distillation method is proposed to handle the long-tailed distribution in the image classification task. The application of KD in MLIC can be found in [Song *et al.*, 2021]. In that work, model distillation is adopted to alleviate the model bias toward difficult categories.

3 Methodology

3.1 Preliminary and Overview

Given an MLIC Task $\mathcal{T} := \{(X, Y)\}$ where X is the image set and Y is its corresponding label set, the goal is to establish a visual learning model $F(\cdot)$, which is able to predict the labels of a given image $x \in X$, $y \leftarrow F(x)$. $y = [y(1), y(2), \cdots, y(m)] \in Y$ is a m-dimensional binary label vector and m is the number of categories. The binary element $y(j) \in \{0, 1\}$ indicates the existence of the corresponding category in an image. Let $\mathcal{C} = \{c_1, c_2, \cdots, c_m\}$ be the category set where $|\mathcal{C}| = m$. With regard to the multi-class classification issue, a larger m implies a high-dimensional label space and thus a more difficult MLIC task. MLIC often suffers from more complexity in comparison to ordinary single-label multi-class image classification even in the same label space because of the label correlation. The divide-and-conquer strategy is an intuitive and common way for tackling such complex tasks. The basic idea of this strategy is to decompose the complex task into a set of simpler sub-tasks, and then assemble the sub-solutions to yield the final solution of the original task.

In this paper, we follow such a strategy and propose a Parallel Self-Distillation (PSD) framework to address MLIC issue. The architecture of PSD is shown in Figure 2. In PSD, the first step is to decompose the original MLIC task $\mathcal{T}$ into

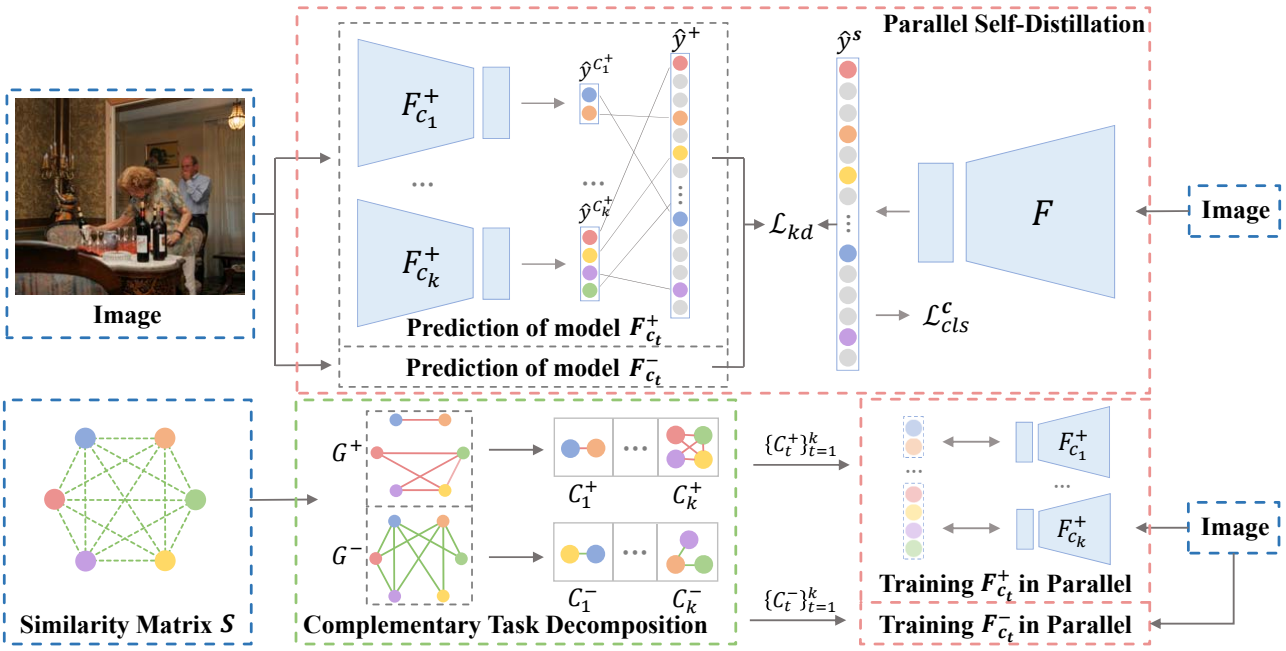

Figure 2: **The overview of the PSD framework.** Blue, green, and red boxes indicate input, task decomposition and the PSD main flow respectively. The superscripts $+$ and $-$ respectively represent co-occurrence and dis-occurrence branches. The omitted operations in the dis-occurrence branch are the same as the ones in co-occurrence branch. Their only difference is their applied sub-tasks.

simpler MLIC sub-tasks by dividing the category set into several smaller subsets with a decomposition strategy $\{T_t\}_{t=1}^k = \Psi(\mathcal{T}, \mathcal{C})$ where k is the number of sub-tasks and $\Psi(\cdot,\cdot)$ is a task decomposition strategy. $T_t := \{(X_{C_t}, Y_{C_t}) | C_t \in \mathcal{C}\}$ is the t-th sub-task where C_t is a subset of $\mathcal{C}$ and is the label space of T_t. X_{C_t} is the images, which contain categories in C_t, and Y_{C_t} is the label set of these corresponding images. The second step is to train an MLIC model $F_{C_t}(\cdot)$ for each sub-task individually. Finally, these trained MLIC models are assembled to yield a compact model which considers them as teachers for knowledge distillation. Moreover, we elaborate two task decomposition strategies named Co-occurrence Graph Partition (CGP) and Dis-occurrence Graph Partition (DGP) for PSD. The decomposition issue is regarded as a spectral clustering problem. In CGP, a label co-occurrence graph is constructed to assign co-occurring labels into the same cluster to induce the model to learn the joint patterns. On the contrary, DGP constructs a label dis-occurrence graph for spectral clustering, and tends to assign labels without co-occurrence into the same sub-task to learn the category-specific patterns. These two strategies are intrinsically complementary with each other and will be applied together to PSD , namely Complementary PSD (CPSD). Notably, the proposed approach can be flexibly plugged into any deep learning-based MLIC models for further improvement.

3.2 Complementary Task Decomposition

Task decomposition is the key of the PSD framework. In the design of the MLIC task composition strategy, two important aspects should be paid attention to. One is the simplification of the task complexity by reducing the dimension of label space. The other is the label correlation exploitation. In model optimization, these two aspects are somewhat conflicted. The label correlation complicates the MLIC task and easily triggers model overfitting to mislead the feature learning to learn features of the co-occurring object instead of the original ones.

In order to avoid this, we design two task decomposition strategies named Co-occurrence Graph Partition (CGP) and Dis-occurrence Graph Partition (DGP) based on spectral clustering. Finally, these two aspects of knowledge will be distilled into a unified MLIC model. CGP and DGP translate the task decomposition issue as the unsupervised category graph partition problem for solution. We construct a *co-occurrence graph* G^+ and a *dis-occurrence graph* G^- to encode the joint and specific patterns among categories respectively. By considering categories $\mathcal{C}$ as vertices and measuring the co-occurrence probabilities of categories as their similarities, we can define a $m \times m$-dimensional similarity matrix S. Its ij-th element is $S_{ij} = e_{ij}/n_i \in [0, 1]$ where e_{ij} is the amount of images containing both c_i and c_j. And n_i is the amount of images containing c_i. Mathematically speaking, let P be the affinity matrix of category graph G. The affinity matrices of G^+ and G^- can be denoted as follows,

$$P = \begin{cases} P^+ = 0.5 \times (\sqrt[\tau]{S} + \sqrt[\tau]{S}^T), & G = G^+ \\ P^- = I - 0.5 \times (\sqrt[\tau]{S} + \sqrt[\tau]{S}^T), & G = G^- \end{cases} \quad (1)$$

where τ is a positive hyper-parameter for smoothing the co-occurrence probabilities which follow the long-tail distribution. A higher value of τ is able to alleviate more suppression of the head to the tail. Since G^+ and G^- are the undirected graphs, we also symmetrize the affinity matrices. The affinity matrices P^+ and P^- respectively encode the degrees of co-occurrence and dis-occurrence among categories. Therefore, we can use them to produce graph Laplacians, and then

conduct spectral clustering to partition the label space as a normalized graph cut problem,

$$\hat{F} \leftarrow \arg\min_{F} \text{Trace}(F^T D^{-\frac{1}{2}} L D^{-\frac{1}{2}} F), \text{ s.t. } F^T F = I, \quad (2)$$

where $L = D - P$ is the Laplacian matrix and D is the degree matrix of G. D is a diagonal matrix whose ii-th element $D_{ii} = \sum_j P_{ij}$. F is the learned graph embedding of vertices (categories), which encode the co-occurrence or dis-occurrence information of each category. Each of its columns encodes a graph cut operation represented by a vector. The aforementioned optimization problem can be efficiently solved by eigenvalue decomposition. The optimal $\hat{F}$ is the eigenvectors corresponding to the top-k minimum eigenvalues. By employing k-means to $\hat{F}$, the categories can be clustered into k category subsets $\{C_t\}_{t=1}^k$ where $\bigcup_{t=1}^k C_t = \mathcal{C}$ and $\bigcap_{t=1}^k C_t = \oslash$. Then the task can be decomposed into k sub-tasks via data sampling according to the clustered category subset, $\{T_t\}_{t=1}^k$. The sub-tasks generated by CGP $\{T_t^+\}_{t=1}^k$ are used to induce the model to learn the features of co-occurring objects. The sub-tasks generated by DGP $\{T_t^-\}_{t=1}^k$ exhibit lower complexity because of the neglect of the label correlation and encourage the model to focus on the extraction of category-specific features. All the teacher models trained by these sub-tasks will finally be used in the Parallel Self-Distillation (PSD), $y \underset{(x,y)\in T_t}{\leftarrow} \sigma(F_{C_t}(x))$ where σ means sigmoid activation, $T_t \in \{T_t^-\}_{t=1}^k \cup \{T_t^+\}_{t=1}^k$ since we expect to reconcile both aspects of knowledge and exploit their complementarity natures for better supervising the training of the student model. Moreover, the teacher models trained by these sub-tasks and generated from both strategies are highly different, which can further benefit PSD from the perspective of ensemble learning.

3.3 Parallel Self-Distillation

For each sub-task T_t, we train a teacher model,

$$\hat{F}_{C_t} \leftarrow \arg\min_{F_{C_t}} L_{cls}^{C_t} \quad (3)$$

with the Asymmetric Loss (ASL) [Ridnik *et al.*, 2021] as the classification loss to suppress the negative effects from superabundant negative samples,

$$L_{cls}^{C_t} = \sum_{x_i \in X_{C_t}} \sum_{c_j \in C_t} \begin{cases} (1-\hat{y}_i^{C_t}(j))^{\gamma_+} \log(\hat{y}_i^{C_t}(j)), y_i(j)=1 \\ \hat{\boldsymbol{y}}_i^{C_t}(j)^{\gamma_-} \log(1-\hat{\boldsymbol{y}}_i^{C_t}(j)), y_i(j)=0 \end{cases} \quad (4)$$

where y_i the ground truth label of x_i and $\hat{y}_i^{C_t} = \sigma(F_{C_t}(x_i))$ is its predictions with respect to the category subset C_t. Its j-th element $\hat{y}_i^{C_t}(j) \in [0,1]$ is the predicted label (probability) of the sample x_i with respect to the category c_j, and $\hat{\boldsymbol{y}}_i^{C_t}(j) = \max(\hat{y}_i^{C_t}(j) - \mu, 0)$. $\mu \geq 0$ is a threshold to filter out the negative samples with low predictions. $\gamma_+ \geq 0$ and $\gamma_- \geq 0$ are respectively the positive and negative focusing hyper-parameters defined in ASL.

Once all teacher models have been obtained, we can merge their prediction results to yield a complete logit prediction according to the category order,

$$\hat{y}_i = \rho(\{\hat{F}_{C_t}(x_i) | x_i \in X_{C_t}\}_{t=1}^k), \quad (5)$$

where $\rho(\cdot)$ is a label merging and reshuffling operation. Here, let $\hat{y}_i^+$ and $\hat{y}_i^-$ be the logit predictions produced by the teacher models based on CGP and DGP respectively. We establish a student model $F(\cdot)$ with the same architecture as teachers to produce the logit predictions of a sample, $\hat{y}_i^{\boldsymbol{s}} = F(x_i)$. The Mean Square Error (MSE) is adopted to measure the discrepancy between the logit predictions of the student model and the teacher models,

$$L_{kd} = \frac{1}{2} \sum_i \{||\hat{y}_i^{\boldsymbol{s}} - \hat{y}_i^+||_2^2 + ||\hat{y}_i^{\boldsymbol{s}} - \hat{y}_i^-||_2^2\}, \quad (6)$$

as the knowledge distillation loss for supervising the student model training.

Finally, the optimal student model can be obtained by addressing the following optimization problem,

$$\hat{F} \leftarrow \arg\min_{F} L_{cls}^{\mathcal{C}} + L_{kd}, \quad (7)$$

where $L_{cls}^{\mathcal{C}}$ is the ASL loss of the original task $\mathcal{T}$. $L_{cls}^{\mathcal{C}}$ can be constructed with Equation 4 based on the full data X. By the above manner, the learned optimal MLIC model $\hat{F}$ will finally incorporate the knowledge acquired by the CGP and DGP-based teacher models, and then we can use it to infer the label of a multi-label image in the testing stage, $y = \sigma(\hat{F}(x))$.

4 Experiments

4.1 Experimental Setup

Datasets. Two widely used MLIC datasets, named MS-COCO and NUS-WIDE, are used for the evaluation of our method. MS-COCO contains 122,218 images with 80 categories of objects in natural scenes, including 82,081 images for training and 40,137 images for validation. In the official partition of NUS-WIDE dataset, it contains 125,449 labeled training pictures and 83,898 labeled test pictures from Flickr, which share 81 labels in total.

Following the conventions, mAP (mean average precision) is deemed as the main evaluation metric. We also report overall precision (OP), recall (OR), F1-measure (OF1) and per-category precision (CP), recall (CR), F1-measure (CF1).

Implementation details. We conduct experiments on three popular backbones, namely ResNet101 [He *et al.*, 2016], TResNetL1 [Ridnik *et al.*, 2021] and ResNeXt50-SWSL [Yalniz *et al.*, 2019], which are all pretrained on ImageNet-1K. A naive Vision Transformer encoder [Dosovitskiy *et al.*, 2020] named ResNet101-TF is implemented with visual tokens extracted from ResNet101. In ResNet101-TF, m class tokens are extracted from GloVe [Pennington *et al.*, 2014] for class predictions, where the depth, number of multi-heads attention, and hidden dimensions are set to be 3, 4, and 1024 respectively. Q2L [Liu *et al.*, 2021] is also adopted to verify the effect of our approach on well-designed methods. All experiments follow a training pipeline where Adam optimizer is used with weight decay of 10^{-4} under a batch size of 32. ASL is applied as the default classification loss function, and the hyper-parameters of ASL are simply left as their default settings. τ in Equation 1 is set to be 3. We set the training epoch to be 20 and 80 for sub-models and the compact global model individually. The number of clusters k will be discussed in our ablation study.

Methods	Backbone	Resolution	mAP	CP	CR	CF1	OP	OR	OF1
ResNet-101 [He *et al.*, 2016]	ResNet101	224×224	78.3	80.2	66.7	72.8	83.9	70.8	76.8
DSDL [Zhou *et al.*, 2021b]	ResNet101	448×448	81.7	84.1	70.4	76.7	85.1	73.9	79.1
CPCL [Zhou *et al.*, 2021a]	ResNet101	448×448	82.8	85.6	71.1	77.6	86.1	74.6	79.9
ML-GCN [Chen *et al.*, 2019c]	ResNet101	448×448	83.0	85.1	72.0	78.0	85.8	75.4	80.3
KSSNet [Liu *et al.*, 2018]	ResNet101	448×448	83.7	84.6	73.2	77.2	87.8	76.2	81.5
MS-CMA [You *et al.*, 2020]	ResNet101	448×448	83.8	82.9	74.4	78.4	84.4	77.9	81.0
MCAR [Gao and Zhou, 2021]	ResNet101	448×448	83.8	85.0	72.1	78.0	88.0	73.9	80.3
Q2L-R101 [Liu *et al.*, 2021]	ResNet101	448×448	84.9	84.8	74.5	79.3	86.6	76.9	81.5
ResNet101*(baseline)	ResNet101	448×448	81.6	80.6	72.7	76.4	83.7	76.7	80.0
Ours + ResNet101	ResNet101	448×448	83.1	83.5	73.6	78.2	84.8	77.3	80.9
ResNet101 + TF*	ResNet101	448×448	84.3	87.4	71.6	78.7	87.9	75.2	81.0
Ours + ResNet101 + TF	ResNet101	448×448	**85.2**	84.9	75.5	**79.9**	85.6	78.5	**81.9**
Q2L-R101*	ResNet101	448×448	84.0	82.0	**75.8**	78.8	83.3	**78.8**	81.0
Ours + Q2L-R101	ResNet101	448×448	84.9	**88.4**	71.7	79.2	**89.3**	74.8	81.4
SSGRL [Chen *et al.*, 2019a]	ResNet101	576×576	83.8	**89.9**	68.5	76.8	**91.3**	70.8	79.7
C-Trans [Lanchantin *et al.*, 2021]	ResNet101	576×576	85.1	86.3	74.3	79.9	87.7	76.5	81.7
ADD-GCN [Ye *et al.*, 2020]	ResNet101	576×576	85.2	84.7	75.9	80.1	84.9	79.4	82.0
Q2L-R101 [Liu *et al.*, 2021]	ResNet101	576×576	86.5	85.8	76.7	81.0	87.0	78.9	82.8
ResNet101 + TF*	ResNet101	576×576	85.9	88.6	73.4	80.3	88.8	76.8	82.4
Ours + ResNet101 + TF	ResNet101	576×576	**86.7**	83.5	**79.0**	**81.2**	84.5	**81.4**	**82.9**
TResL [Ridnik *et al.*, 2021]	TResNetL	448×448	86.6	87.2	76.4	81.4	88.2	79.2	81.8
Q2L-TResL [Liu *et al.*, 2021]	TResNetL	448×448	**87.3**	**87.6**	76.5	81.6	**88.4**	79.2	81.8
TResL*(baseline)	TResNetL	448×448	86.2	85.0	77.5	81.1	85.6	80.4	82.9
Ours + TResL	TResNetL	448×448	**87.3**	85.5	**78.9**	**82.1**	85.7	**81.5**	**83.7**
ML-GCN [Nguyen *et al.*, 2021]	ResNeXt50-SWSL	448×448	86.2	85.8	77.3	81.3	86.2	79.7	82.8
MGTN [Nguyen *et al.*, 2021]	ResNeXt50-SWSL	448×448	87.0	86.1	77.9	81.8	87.7	79.4	83.4
ResNeXt50*(baseline)	ResNeXt50-SWSL	448×448	86.7	85.8	77.8	81.6	86.9	80.3	83.5
Ours + ResNeXt50	ResNeXt50-SWSL	448×448	**87.7**	**86.9**	**78.6**	**82.5**	**87.6**	**80.9**	**84.1**

Table 1: **The MLIC performances of different methods on MS-COCO** with pretrained backbones on ImageNet-1k. * indicates the results reproduced by the corresponding released codes or their modified versions. The best results for each backbone are in **bold.**

4.2 Comparison with State-of-The-Art Methods

Tables 1 and 2 report the MLIC performances of several methods evaluated on MS-COCO and NUS-WIDE datasets respectively. We apply our proposed framework on recent benchmarks to evaluate the effectiveness.

The observations show that our method generally boosts all baselines, and performs the best on both datasets. For example, the enhanced ResNet101, ResNet101-TF, Q2L-R101, TResL and ResNeXt50 get 1.5%, 0.9%, 0.9%, 1.1% and 1.0% gains respectively in mAP over their original ones on MS-COCO dataset. Such gains of ResNet101+TF and TResL are 1.7% and 1.8% on NUS-WIDE, which is a larger scale dataset. These experimental results also imply that our method performs much better on a larger-scale dataset.

In addition, the observations also imply that, based on our framework, the naive model is able to achieve state-of-the-art performances without involving additional costs in parameter scale or more complicated architectures. For example, our method gets 0.7% gains in mAP over MGTN using only its backbone on MS-COCO dataset. Another interesting phenomenon is that we achieve smaller performance gains on the more advanced models. For example, CPSD improves more on ResNet101 in comparison with ResNet101-TF and Q2L-R101. We attribute this to the fact that the more powerful approaches are much harder to trap in model overfitting. Even so, our method still introduces a considerable improvement.

Methods	mAP	CF1	OF1
MS-CMA [You *et al.*, 2020]	61.4	60.5	73.8
SRN [Zhu *et al.*, 2017]	62.0	58.5	73.4
CPCL [Zhou *et al.*, 2021a]	62.3	59.2	73.0
CADM [Chen *et al.*, 2019b]	62.8	60.7	74.1
Q2L-R101 [Liu *et al.*, 2021]	65.0	63.1	75.0
ResNet101+TF*	64.1	62.8	74.9
Ours+ResNet101+TF	**65.8**	**64.0**	**75.3**
TResL [Ridnik *et al.*, 2021]	65.2	63.6	75.0
Q2L-TResL [Liu *et al.*, 2021]	66.3	64.0	75.0
TResL*(baseline)	64.7	63.7	75.0
Ours+TResL	**66.5**	**64.6**	**75.5**

Table 2: **The MLIC performances of different methods on NUS-WIDE** with pretrained backbones on ImageNet-1k where the image resolution is 448×448.* indicates the results reproduced by the corresponding released codes or their modified versions. The best results are in **bold.**

4.3 Ablation Study

Discussion on task decomposition strategy. We plot the performances of PSD under different k with different task decomposition strategies Ψ on both MS-COCO and NUS-WIDE in Figure 3. The results indicate that the performances of PSD increased along with the increase of k with all strategies on both datasets. Moreover, our proposed strategies con-

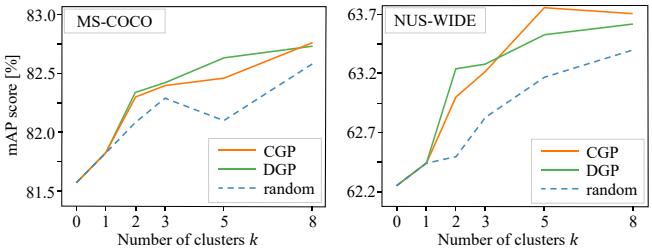

Figure 3: **The performances of ResNet101 enhanced by PSD under different k with different strategies on both MS-COCO and NUS-WIDE.** CGPD, DGPD mean Co/Dis-Occurrence Graph Partition Distillation, which indicate student models that use only CGP or DGP decomposition strategy. 0 and 1 respectively represent the baseline and the baseline with self-distillation. We conduct 3 independent experiments on random strategy and report the mean of their results for eliminating the effects of randomness.

	R101	TResL	ResX50	Q2L-R101	R101+TF
baseline	81.6	86.2	86.7	84.0	84.3
+ SD	81.9	86.5	86.9	84.2	84.4
+ CGPD	82.4	86.6	87.3	84.6	84.7
+ DGPD	82.7	86.8	87.4	84.3	84.5
+ CPSD	83.1	87.3	87.7	84.9	85.2

Table 3: **Component ablation study of CPSD.** SD, CGPD, DGPD, CPSD mean Self-Distillation, Co/Dis-Occurrence Graph Partition Distillation and Complementary Parallel Self-Distillation respectively. Here, $k = 5$.

sistently perform much better than the random ones when $k \geq 2$. DGPD performs better than CGPD when k is small while the best performances of the two strategies are highly similar. We attribute this to the fact that overemphasizing the label correlation causes model overfitting more easily when the label space is in high-dimensional, while the label correlation is still able to benefit MLIC when the category-specific features are well learned. Actually, reducing the size of the cluster (increasing k) can also be deemed as a natural way to break down label correlation. A larger k means more teacher models are needed to be trained, which leads to higher time cost. To achieve the tradeoff between the performance and the model training time, we set $k = 5$.

Ablation study on components. Table 3 shows the ablation study results of our method with different baselines on MS-COCO. The results show that our PSD framework boosts all the baselines and outperforms the common Self-Distillation (SD) with considerable advantages using CGP, DGP or their combination. For example, CPSD further improves SD by 1.4%, 0.8%, 0.8%, 0.7% and 0.8% on ResNet101, TRestL, ResNeXt50, Q2L-R101 and ResNext101-TF respectively. Moreover, CPSD performs much better than models using only one decomposition strategy, i.e. CGPD and DGPD. The improvements under different baselines are around 0.5%. These observations confirm the effectiveness of our method.

4.4 Explainable Visualization Study

We apply Class Activation Map (CAM) on the validation set of MS-COCO to visualize the implicit attentions of different

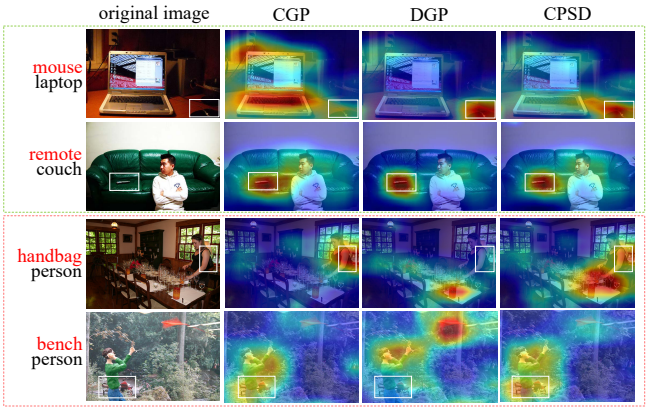

Figure 4: **Activation map visualizations of images under different decomposition strategies.** *CGP, DGP* columns are the activation map visualizations of corresponding sub-model $F^{+}_{C_t}$ and $F^{-}_{C_t}$. *CPSD* column is the visualizations of our final model. The red label indicates the target class we want to activate, while the others indicate the co-occurring classes.

images with respect to a specific category using different task decomposition strategies as shown in Figure 4. In this figure, the columns from left to right respectively show the original images and the images highlighted by the activations maps of the baseline ResNet101 models enhanced with PSD under CGP, DGP and their combination (CPSD). The first two cases (the top two rows) show that the category-specific features have been ignored by the model based on CGP due to the model overfitting caused by the overemphasis of the label correlation, while they are well learned by our DGP models. The last two cases show that the label co-occurrence information has not been exploited by the model based on DGP due to the occlusion, however the co-occurring categories are able to assist the model in discovering these features with the help of the label correlation information. The visualizations demonstrate that our proposed method can perform well in each of these situations and also reflect that our method can better exploit and reconcile the category-specific knowledge and label correlation knowledge.

5 Conclusion

In this paper, we proposed a simple yet effective Parallel Self-Distillation (PSD) framework in which the original complex MLIC task is decomposed into a set of simpler sub-tasks via label partition. Then multiple teacher models are trained in parallel to address these sub-tasks individually. The final model is obtained through the ensemble of these teacher models with knowledge distillation. For better boosting PSD, we introduce two task decomposition strategies, which address the task decomposition issue through conducting two complementary co-occurrence graph partitions. These two strategies, which respectively induce the models to learn the category-specific and category-correlated knowledge, are applied to PSD to set up the sub-tasks. Extensive experimental results on MS-COCO and NUS-WIDE demonstrate that our framework is effective and can be plugged into different approaches to boost the performances.

Acknowledgements

Reported research is partly supported by the National Natural Science Foundation of China under Grant 62176030 and 71621002, the Natural Science Foundation of Chongqing under Grant cstc2021jcyj-msxmX0568, and the Strategic Priority Research Program of Chinese Academy of Sciences Grant XDA27030100.

References

[Chen *et al.*, 2019a] Tianshui Chen, Muxin Xu, Xiaolu Hui, Hefeng Wu, and Liang Lin. Learning semantic-specific graph representation for multi-label image recognition. In *ICCV*, pages 522–531, 2019.

[Chen *et al.*, 2019b] Zhao-Min Chen, Xiu-Shen Wei, Xin Jin, and Yanwen Guo. Multi-label image recognition with joint class-aware map disentangling and label correlation embedding. In *IEEE ICME*, pages 622–627. IEEE, 2019.

[Chen *et al.*, 2019c] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *CVPR*, pages 5177–5186, 2019.

[Chen *et al.*, 2021] Yawen Chen, Zeyi Wen, Bingsheng He, and Jian Chen. Efficient decomposition selection for multi-class classification. *IEEE TKDE*, 2021.

[Dosovitskiy *et al.*, 2020] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[Galar *et al.*, 2011] Mikel Galar, Alberto Fernández, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes. *Pattern Recognition*, 44(8):1761–1776, 2011.

[Gao and Zhou, 2021] Bin-Bin Gao and Hong-Yu Zhou. Learning to discover multi-class attentional regions for multi-label image recognition. *IEEE TIP*, 30:5920–5932, 2021.

[Gou *et al.*, 2021] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *IJCV*, 129(6):1789–1819, 2021.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Hinton *et al.*, 2015] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[Lanchantin *et al.*, 2021] Jack Lanchantin, Tianlu Wang, Vicente Ordonez, and Yanjun Qi. General multi-label image classification with transformers. In *CVPR*, pages 16478–16488, 2021.

[Liu *et al.*, 2018] Yongcheng Liu, Lu Sheng, Jing Shao, Junjie Yan, Shiming Xiang, and Chunhong Pan. Multi-label image classification via knowledge distillation from weakly-supervised detection. In *ACM MM*, pages 700–708, 2018.

[Liu *et al.*, 2021] Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Query2label: A simple transformer way to multi-label classification. *arXiv preprint arXiv:2107.10834*, 2021.

[Nguyen *et al.*, 2021] Hoang D Nguyen, Xuan-Son Vu, and Duc-Trong Le. Modular graph transformer networks for multi-label image classification. In *AAAI*, volume 35, pages 9092–9100, 2021.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[Ridnik *et al.*, 2021] Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. Asymmetric loss for multi-label classification. In *ICCV*, pages 82–91, 2021.

[Song *et al.*, 2021] Liangchen Song, Jialian Wu, Ming Yang, Qian Zhang, Yuan Li, and Junsong Yuan. Handling difficult labels for multi-label image classification via uncertainty distillation. In *ACM MM*, pages 2410–2419, 2021.

[Wang *et al.*, 2016] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *CVPR*, pages 2285–2294, 2016.

[Wu *et al.*, 2020] Tong Wu, Qingqiu Huang, Ziwei Liu, Yu Wang, and Dahua Lin. Distribution-balanced loss for multi-label classification in long-tailed datasets. In *ECCV*, pages 162–178. Springer, 2020.

[Xiang *et al.*, 2020] Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *ECCV*, pages 247–263. Springer, 2020.

[Yalniz *et al.*, 2019] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*, 2019.

[Ye *et al.*, 2020] Jin Ye, Junjun He, Xiaojiang Peng, Wenhao Wu, and Yu Qiao. Attention-driven dynamic graph convolutional network for multi-label image recognition. In *ECCV*, pages 649–665. Springer, 2020.

[You *et al.*, 2020] Renchun You, Zhiyao Guo, Lei Cui, Xiang Long, Yingze Bao, and Shilei Wen. Cross-modality attention with semantic graph embedding for multi-label classification. In *AAAI*, volume 34, pages 12709–12716, 2020.

[Yuan *et al.*, 2020] Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *CVPR*, 2020.

[Zhang *et al.*, 2019] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pages 3713–3722, 2019.

[Zhou *et al.*, 2021a] Fengtao Zhou, Sheng Huang, Bo Liu, and Dan Yang. Multi-label image classification via category prototype compositional learning. *IEEE TCSVT*, 2021.

[Zhou *et al.*, 2021b] Fengtao Zhou, Sheng Huang, and Yun Xing. Deep semantic dictionary learning for multi-label image classification. In *AAAI*, volume 35, pages 3572–3580, 2021.

[Zhou *et al.*, 2021c] Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *arXiv preprint arXiv:2102.00650*, 2021.

[Zhu *et al.*, 2017] Feng Zhu, Hongsheng Li, Wanli Ouyang, Nenghai Yu, and Xiaogang Wang. Learning spatial regularization with image-level supervisions for multi-label image classification. In *CVPR*, pages 5513–5522, 2017.