Multi-label Image Recognition with Asymmetric Co-occurrence Dependency Graphs

1st Yuhang Qi
Beijing Jiaotong University
Beijing, China
18120118@bjtu.edu.cn

2nd Yuchun Guo
Beijing Jiaotong University
Beijing, China
ychguo@bjtu.edu.cn

3rd Yishuai Chen
Beijing Jiaotong University
Beijing, China
yschen@bjtu.edu.cn

***Abstract*—Multi-label image recognition is a practical and challenging task. Modeling co-occurrence dependencies between categories is the key to improve performance. Existing methods use conditional probability to measure co-occurrence dependencies, and represent co-occurrence dependencies among all categories in the form of directed graph. Then the Graph Convolution Network (GCN) is applied on the directed graph to transfer dependent category features along the edge direction. However, the occurrence frequencies of different categories are different. Accordingly, the conditional probabilities between a pair of common and rare categories are highly asymmetric so that most rare categories have no in-edges to receive transferred knowledges from other categories. Therefore, this paper investigates the effects of edge direction between two co-occurred categories on the recognition performance, then proposes a model to work on a pair of directed graphs to learn a comprehensive representation of co-occurrence dependency. Extensive experiments on public benchmarks show that our method can achieve better performance than baseline models. On some multi-label image recognition datasets with strong co-occurrence dependencies, our method can improve the mAP by 4%.**

***Index Terms*—multi-label, image, co-occurrence, graph**

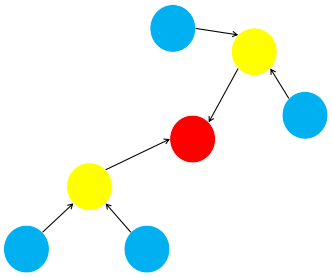
I. Introduction

Multi-label image recognition is a practical task in computer vision. Its goal is to predict the set of categories in an image. It can be applied to many downstream tasks, such as pedestrian attribute recognition [8], recommendation system [9], etc. Compared with the general image classification task, some categories in multi-label image recognition depend on each other, and these categories normally co-occur in an image. Modeling this co-occurrence dependencies is the key to improve the performance of multi-label image recognition.

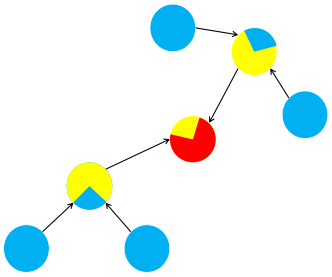
The early method of multi-label image recognition is to deal with categories independently, then the multi-label image recognition is decomposed into several general binary classification tasks, each task only recognizes one category. With the development of deep convolution neural network [1], [10]–[12], the performance of this method is also improved. In recent years, many researches have improved the performance of multi-label image recognition by modeling the dependencies between categories. The method [13] based on recurrent neural network (RNN) [24], [25] transforms the target categories set into a sequence for prediction. The methods [14], [15] based on attention mechanism [3], [4] improve the performance by obtaining the feature map information related to specific categories on a single image. The method [16] models the dependencies between categories by Graph Convolutional Networks (GCN) [17] to further improve the performance of multi-label image recognition.

Using GCN to model co-occurrence dependencies has achieved competitive performance, but this method still has some shortcomings. In [16], the researchers used conditional probabilities to quantify the co-occurrence dependencies, and represented the co-occurrence dependencies in the form of a directed graph. In directed graph, nodes represent categories and directed edges represent co-occurrence dependencies, if node A has an edge that pointed to node B, it means that under the condition that category A appears, the category B is very likely to appear. The GCN [17] is applied to this graph, and a vector representation is output for each node as the classifier vector of the corresponding category. The forward of GCN can be summarized as the aggregation and transformation of node features. In the feature aggregation process of directed graph, the direction of the edge represents the direction of the node feature aggregation, if node A has an edge that points to the node B, then node B will aggregate the feature of node A into its own feature, but on the contrary, node A will not aggregate the feature of node B. The visualization process is shown in Fig. 1. This ability to propagate information on the graph through the aggregation of node features enables GCN to model the graph-structure data. So the directions of edges are important for modeling the co-occurrence dependencies. However, there are two problems with this method.

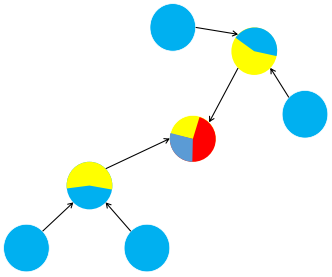
- In the graph constructed by [16], many edges are from the low frequency category to the high frequency category. For example, in the MS-COCO [18] dataset, 'person' is the most frequent category, with 55 nodes having edges pointing to the 'person' node. This means that the features of 'person' are enhanced with 55 low frequency categories. However, these low frequency categories did not transfer features from the high frequency categories. This will lead to insufficient information of low-frequency categories learned by the model, which will lead to poor performance.
- The second problem is that in many co-occurrence dependency graphs built from benchmark datasets, many nodes do not have edges pointing to themselves from other



(a)

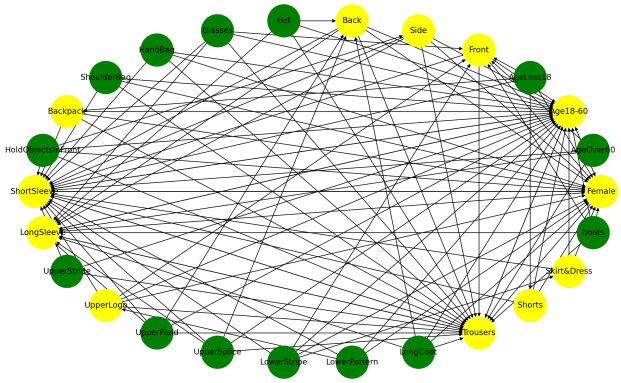


(b)

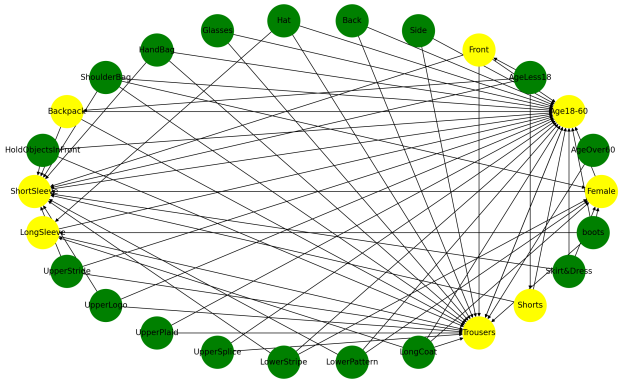


(c)

Fig. 1. The forward propagation of GCN. (a) Initial state. Yellow nodes have edges pointing to red node. Blue nodes have edges pointing to yellow nodes. (b) After one time of feature aggregation, red node aggregates the features of yellow nodes, but yellow nodes does not aggregate the features of red node. (c) After two times of feature aggregation, the red node can get the features of the farther neighbor nodes(blue nodes). This is the key of modeling graph structure data.



(a)



(b)

Fig. 2. Two directed graphs constructed from PA-100K dataset. The yellow nodes have edges that point to themselves from other nodes. The green nodes do not have edges that point to themselves from other nodes. (a)An edge is defined when the conditional probability is greater than 0.4. (b)An edge is defined when the conditional probability is greater than 0.6. As can be seen from (a) and (b), most nodes do not have edges pointed to themselves from other nodes.

nodes. In other words, these nodes only perform feature transformation during the forward of GCN, but not feature aggregation. Only performing feature transformation is equivalent to general feed-forward neural network, which greatly limits the ability of GCN. For example, there are 57 nodes in the directed graph constructed by these 80 categories on MS-COCO [18] do not have edges pointing to themselves from other nodes. In other benchmark such as PA-100K [19], the same phenomenon happens. Among the 26 nodes of PA-100K, 14 nodes do not have edges pointing to themselves from other nodes. The directed graph constructed from PA-100K dataset is in Fig. 2.

For the first problem, an intuitive method is to reverse the direction of the edge, so that many aggregation directions will be from the category with higher frequency to the category with lower frequency. In other words, when the conditional probability $P(B \mid A)$ is greater than a certain threshold, an edge will be defined from B to A, rather than an edge from

A to B as in the baseline model. This paper evaluates the effect of defining the direction of the edge in this way on the performance through ablation studies. Experimental results on benchmark datasets show that the performance of defining direction in this way is better than that of baseline model. The reason for this is that the category with higher frequency makes information supplement for the category with lower frequency. Using the features of the high frequency categories to enhance the features of the low frequency categories can improve the recognition performance of the low frequency categories. The second problem can also be alleviated by using this way. After changing the directions of edges, most nodes will have edges pointing to themselves from other nodes. For example, in the MS-COCO [18], only 15 nodes do not have edges pointing to themselves from other nodes, but in the original graph, there are 57 nodes do not have edges pointing to themselves from other nodes. In PA-100K [19], all nodes have edges pointing to themselves from other nodes.

Using two graphs at the same time, which are constructed by the baseline model and our method of defining the direction of edges, can further solve the second problem and improve the performance. In order to improve the flexibility of the model, two different conditional probability thresholds are used to create a pair of asymmetric graphs. GCNs are used to extract node features respectively. Then the outputs of GCNs are fused as the final classifiers. Experimental results on two benchmark datasets show that our method can improve the performance of multi-label image recognition model. On some multi-label image recognition datasets with strong co-occurrence dependencies such as PA-100K [19], our method can improve the mAP by 4%.

II. Related Work

Multi-label image recognition is an extension of image classification. Compared with image classification task, each image in multi-label image recognition has multiple categories. There are co-occurrence dependencies between these categories. Many researches are devoted to improving the performance of multi-label image recognition by using co-occurrence dependencies between categories.

Wang et al. [13] proposed to use RNN [24], [25] to model the dependencies between categories. They converted multi-label image recognition into predicting a category sequence in the image. However, it is not accurate to use serialization relation to generalize the dependencies relationship between categories. In fact, co-occurrence dependency is a pairwise asymmetric relationship rather than a linear chain relationship.

Zhu et al. [14] optimized the performance of multi-label image recognition from another perspective. They used the attention mechanism [3], [4] to find the feature map regions associated with different categories from the image feature map, then extracted the feature map information strongly related to the categories for recognition. This method does not directly use global statistical information between categories. Wang et al. [15] combined RNN with attention mechanism, but this method still can not accurately model co-occurrence dependency.

Chen et al. [16] proposed to quantify the co-occurrence dependencies between categories using the conditional probabilities obtained by statistics, and represented them in the form of directed graph, then used GCN [17] to model the co-occurrence dependencies. This method fails to supplement the information of the low frequency categories. In addition, a large number of nodes in the co-occurrence dependency graph do not have edges pointing to themselves from other nodes which greatly limits the ability of GCN.

III. Approach

Firstly, this section gives an overall description of our model. Our model is divided into two parts, full convolution neural network part and asymmetric graphs modeling part. The full convolution neural network (CNN) is used to extract the image feature map $F \in R^{H \times W \times C}$, then the image feature map is compressed into a feature vector $f \in R^C$ by global maximum pooling (GMP) or global average pooling (GAP). Where H, W and C represent the height, width and channel of the feature map respectively. In the part of asymmetric graphs modeling, node features of two graphs constructed by two ways to define the direction of an edge are extracted by two GCNs, and then fused into a category classifier $W = \{w_i\}_{i=1}^{N}$, where N is the number of categories. Finally, the image feature f will do the dot product with the classifier w_i of each category, and the occurrence probability of objects belonging to category i can be obtained through applying sigmoid function to the dot product result. The overall framework is in Fig 3.

A. Co-occurrence Dependency Graph

In multi-label image recognition, many categories do not appear independently, but have co-occurrence dependencies. One way to measure this co-occurrence dependency is to use conditional probability [16]. For example, there are labels L_i and L_j in the dataset. $P(L_i \mid L_j)$ indicates the probability of occurrence of label L_i when label L_j appears. The appearing times of label L_i and L_j can be obtained from statistics on the training dataset. At the same time, the appearing times of the labels pair (i, j) can also be obtained. Then the conditional probability can be evaluated by the following formula:

$$P(L_i \mid L_j) = M_{ij}/M_j \tag{1}$$

$$P(L_j \mid L_i) = M_{ij}/M_i \tag{2}$$

where M_i, M_j are the appearing times of label L_i and L_j, M_{ij} is the appearing times of labels pair (i, j).

Then two co-occurrence dependency graphs can be constructed according to the conditional probability. Two co-occurrence graphs are directed graphs. In both directed graphs, nodes are labels. The two graphs are diametrically opposite in the direction of edge. A intuitive method is to define an edge pointing from L_j to L_i on the first graph and an edge pointing from L_i to L_j on the second graph if the conditional

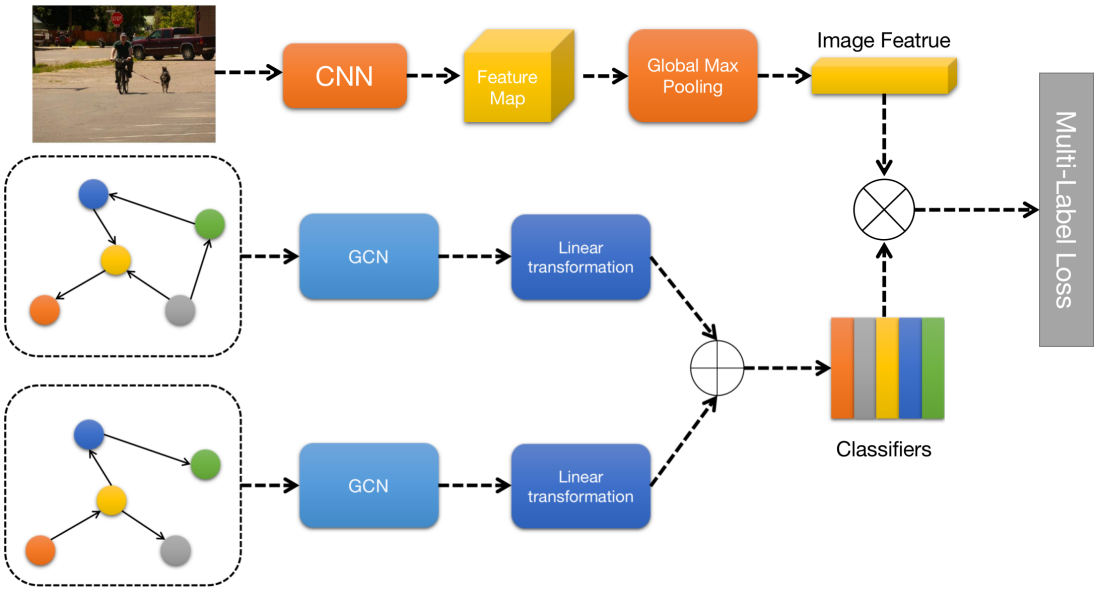


Fig. 3. Overall framework of our model. Our model is divided into two parts, full convolution neural network part and asymmetric graphs modeling part. In asymmetric graphs modeling part, there are two graphs. The two graphs have the same nodes and many edges with opposite directions. However, the first graph has some edges but the second graph does not. This is because the thresholds used to build the two graphs are different. For example, in the first graph, there is an edge from the gray node to the green node, but in the second graph, there is no edge from the green node to the gray node. That's because of $P\left(green \mid gray\right) >= \tau_1$ and $P\left(green \mid gray\right) < \tau_2$. That is what 'asymmetric' means.

probability $P\left(L_i \mid L_j\right) \neq 0$. However, there is a problem with this method. If $P\left(L_i \mid L_j\right) \neq 0$, that means $M_{ij} > 0$, so $P\left(L_j \mid L_i\right) \neq 0$. In other words, for any graph, if there is an edge between two nodes, there must be an another edge with different direction. This will obliterate the directivity of the directed graph and result in the two constructed graphs being exactly the same. In addition, since the conditional probability is obtained by statistical method, there is noise in conditional probability. Some minimal but not zero conditional probabilities may be noise. The statistical results are in Fig 4. In fact, most of the probability values are very small. These probabilities are caused by noise and can not reflect the co-occurrence dependencies between categories. Therefore, in order to solve this problem, as in [16], a threshold τ is introduced to filter conditional probability. The method is to define an edge pointing from L_j to L_i on the first graph and an edge pointing from L_i to L_j on the second graph if the conditional probability $P\left(L_i \mid L_j\right) \geq \tau$. In this way, the direction of edges in the two graphs is completely opposite, and the directions of node feature aggregation in the two graphs are symmetric.

In order to increase the flexibility of the model, two threshold parameters τ_1 and τ_2 are introduced. An edge pointing from L_j to L_i was defined on the first graph when $P\left(L_i \mid L_j\right) \geq \tau_1$. An edge pointing from L_i to L_j is defined on the second graph when $P\left(L_i \mid L_j\right) \geq \tau_2$. In this way, the influence of noise can be eliminated and the structure of the two graphs can be diversified. Due to different thresholds, the aggregation directions of node features in the two graphs are asymmetric. An example are shown in Fig 3.

B. Graph Convolution Network

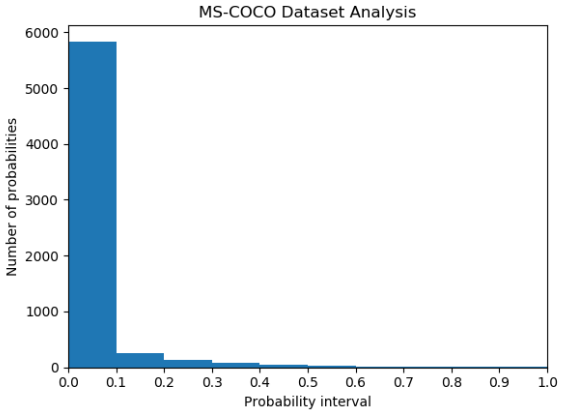
GCN [17] was originally proposed for semi-supervised classification on graph structured data. The forward propagation process of GCN can be summarized as aggregation and transformation of node features. Node feature aggregation enables the information of nodes on the graph to be propagated to other nodes. Each node can perceive the global information of the graph through the topological structure of the graph, so as to enrich the node feature.

GCN takes graph as input. The information of a graph is divided into two parts, one is the node features of the graph, the other is the topology of the graph. Graph node features are generally represented by scalars or vectors. In multi-label image recognition, the 300-dimensional GloVe word embeddings [20] of the labels corresponding to the nodes are used as the node features. For the categories whose names contain multiple words, the label representation are the average of embeddings for all words. The topological structure of a graph is represented by its adjacency matrix $A \in \{0,1\}^{N \times N}$. In a directed graph, $A_{ij} = 1$ indicates that there is an edge from node j to node i. The forward propagation process of GCN can be calculated by the following formula:

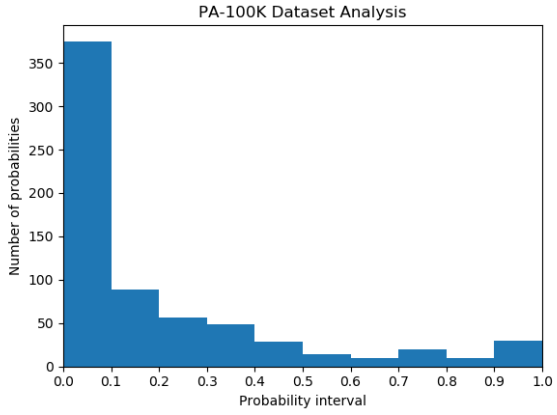
$$Z = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} X \Theta \tag{3}$$

where $\tilde{A} = A + I_N$, I_N is the identity matrix. $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ is the degree matrix of the graph. X is the node feature matrix. $\Theta \in R^{d_{\text{input}} \times d_{\text{output}}}$ is the transformation matrix, d_{input} and d_{output} are the dimensions of input and output, respectively. The formula (3) can abbreviated as follows:

$$\mathrm{Z} = \widehat{\mathrm{A}} \mathrm{X} \Theta \tag{4}$$



(a)



(b)

Fig. 4. Statistical results on (a) MS-COCO and (b) PA-100K. The vast majority of probability values are minimal, and these minimum non-zero probabilities are caused by noise.

$\widehat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$ is called *normalized adjacency matrix*. The aggregation process of node features is implied in matrix multiplication $\widehat{A}X$. Let $M = \widehat{A}X$, and the i-th row of the matrix M represents the feature of the i-th node feature after feature aggregation. It can be regarded as the weighted average of the features of the i-th node and its neighbors, where the weight is given by the i-th row of the *normalized adjacency matrix* $\widehat{A}$. It can be seen from the above formula and Fig. 1 that nodes can aggregate the features from the farther neighbor nodes by stacking GCN layers, then obtain the global information of the graph to enhance the features.

GCNs are used to get node features from the two co-occurrence dependency graphs proposed in the previous section. For each co-occurrence dependency graph, a two-layer GCN is used to get node features. LeakyReLU [21] with the negative slope of 0.2 is adopted as the non-linear activation function after GCN layer. The reason for using only two layers is that with the growing of the layers, the GCN will encounter the problem of over smoothing [22]. The problem of over smoothing can lead to the consistency of node features and make them indistinguishable, which leads to performance degradation. Finally, the node features of the two GCNs will be fused and used as classifiers. Linear transformation is used to map node features to classifier parameter space:

$$W = Z_1 W_1 + Z_2 W_2 \tag{5}$$

Z_1 and Z_2 are the outputs of the GCNs, and W_1 and W_2 are the transformation matrices. For each category, the classifier is a C-dimensional vector, C is also the dimension of image feature vector.

C. Convolution Neural Network

Because of its strong ability to extract spatial local features, CNN [1], [10]–[12] has achieved great success in image classification. In the previous research of multi-label image recognition, ResNet101 [11] is used as the backbone network. For fair comparison, ResNet101 is used as the backbone network on MS-COCO [18] dataset. In order to prove the effectiveness of our method, not only ResNet101 is taken as the backbone network, but also ResNet50 [11] is taken as the backbone network. On PA-100K [19] dataset, ResNet50 is used as the backbone network. ResNet50 and ResNet101 are pre-trained on ImageNet [2] dataset.

IV. Experiment

This section first describes the evaluation metrics and hyper-parameters settings, then reports the experimental results on MS-COCO [18], PA-100K [19] datasets. Finally, ablation studies are presented.

A. Evaluation Metrics

To fairly compare with existing methods [13]–[16], the mean average precision (mAP), average per-class precision (CP), recall (CR), F1 (CF1) and the average overall precision (OP), recall (OR), F1 (OF1) are reported for performance evaluation. For each image, the labels are predicted as positive if the confidences of them are greater than 0.5. The results of top-3 labels are also reported for fairly comparisons. Generally, average overall F1 (**OF1**), average per-class F1 (**CF1**) and **mAP** are relatively more important for performance evaluation.

B. Hyper-parameters Settings

As in [16], the output dimensions of our two-layer GCN are 1024 and 2048, respectively. Since the first co-occurrence dependency graph is the same as that in [16], τ_1 will be set to 0.4 for fair comparison. Competitive performance can be achieved by setting τ_2 between 0.4 and 0.6. And the performance will be reported when τ_2 is set to 0.6. During training, the input images are random cropped and resized into 448 × 448 with random horizontal flips for data augmentation. For network optimization, SGD is used as the optimizer. The momentum is set to be 0.9. Weight decay is 10^{-4}. Cosine annealing learning rate scheduler proposed in [23] are used as learning rate scheduler. The initial learning rate is 0.01 and the network is trained for 100 epochs in total. Without additional

stated, both baseline model and our model are tested in this configuration.

C. Experimental Results

1) MS-COCO: Microsoft COCO [18] is originally constructed for object detection, and it has been adopted to evaluate multi-label image recognition recently. It contains 82,081 images as the training set and 40,504 images as the validation set. There are 80 categories in the dataset. Since the ground-truth labels of the test set are not available, validation set is used to evaluate the performance of all methods. Quantitative results are reported in Table I. The performance of other methods are reported, including CNN-RNN [13], RNN-Attention [15], Order-Free RNN [7], ML-ZSL [6], SRN [14], Multi-Evidence [5], ML-GCN [16], etc. From Table I, our method is better than the baseline methods in all important metrics, whether it is using ResNet50 or ResNet101 as the backbone network.

2) PA-100K: The PA-100K [19] dataset is constructed by images captured from 598 real outdoor surveillance cameras, it includes 100000 pedestrian images. The whole dataset is randomly split into training, validation and test sets with a ratio of 8:1:1. Every image in this dataset was labelled by 26 attributes. ResNet50 are used as the backbone network. The evaluation results are shown in Table II. Our method has a significant improvement over ML-GCN [16]. The significant performance improvement is in line with our expectations because of the stronger co-occurrence dependencies between categories in PA-100K dataset compared to MS-COCO dataset. The categories in MS-COCO dataset are derived from natural scenes, while the categories in PA-100K dataset are pedestrian attributes, which have stronger co-occurrence dependencies. From the perspective of probability, it can be seen that, in the MS-COCO dataset, the average of all conditional probabilities greater than 0.4 is 0.58, while in PA-100K dataset, the average of all conditional probabilities greater than 0.4 is 0.70. This shows that the average co-occurrence dependency on PA-100K dataset is stronger when co-occurrence dependency exists between two categories. The experimental results show that our method makes full use of the information about co-occurrence dependency.

D. Ablation Studies

This section presents the results of the ablation studies. The purpose of our ablation studies is to explore the effect of two ways to define the direction of an edge on performance. The model using first direction is the same as the ML-GCN [16]. For the model using only the second direction, we only need to change the directions of edges of the ML-GCN. The evaluation results on MS-COCO and PA-100K are shown in Table III, Table IV and Table V. On the MS-COCO dataset, the performance of the second direction is slightly better than that of the first one. On PA-100K dataset, the performance of the second direction is significantly better than that of the first one. However, no matter which direction is used, its performance is not as good as the unified model which uses both directions at the same time.

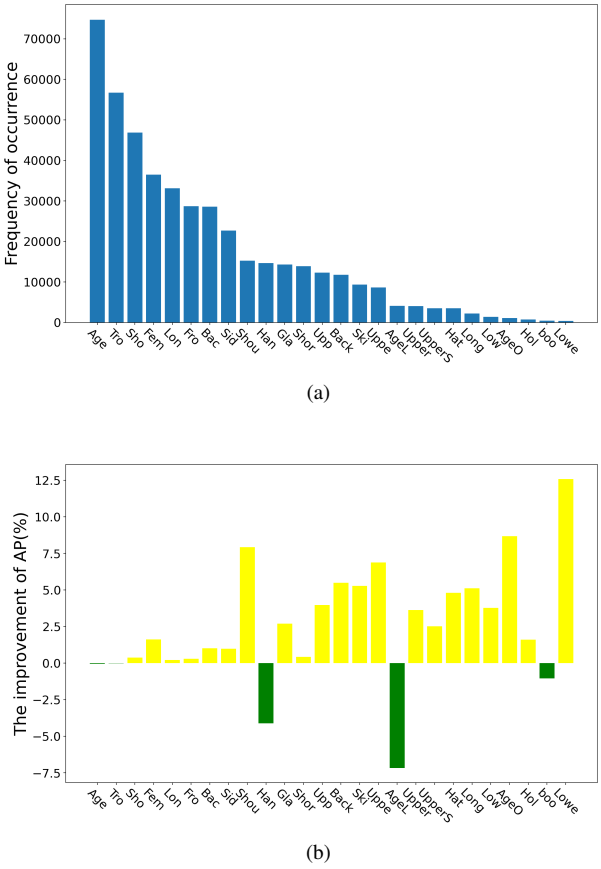


Fig. 5. (a) The frequency of each category. (b) Performance improvement. Performance improvement is measured by average precision (**AP**). The yellow column represents an improvement in performance, and the green column represents a decrease in performance. The categories of the two bar graphs are the same. It can be seen from (a) and (b) that the recognition performance of most low frequency categories is significantly improved.

On the PA-100K dataset, we further analyze the performance improvement of the second way to define the direction of the edge for the categories with higher frequency and the categories with lower frequency. The results are shown in the Fig. 5. From Fig. 5, it is that using the second way to define the direction of the edge can improve the recognition performance of most of the low frequency categories. However, it can be seen that the recognition performance of several categories decreased. This shows that using the second way to define the direction of the edge is also inaccurate for these categories. In other words, each of the two ways has its own advantages. Different ways are suitable for different categories. But in terms of overall performance, it is better to use the second way to define the direction of the edge.

V. Conclusion

In multi-label image recognition, it is very important to model the co-occurrence dependencies between categories

TABLE I
EXPERIMENTAL RESULTS ON MS-COCO

Methods	All							Top-3					
	mAP	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***
CNN-RNN [13]	61.2	-	-	-	-	-	-	66.0	55.6	60.4	69.2	66.4	67.8
RNN-Attention [15]	-	-	-	-	-	-	-	79.1	58.7	67.4	80.4	63.0	72.0
Order-Free RNN [7]	-	-	-	-	-	-	-	71.6	54.8	62.1	74.2	62.2	67.7
ML-ZSL [6]	-	-	-	-	-	-	-	74.1	64.5	69.0	-	-	-
SRN [14]	77.1	81.6	65.4	71.2	82.7	68.9	75.8	85.2	58.8	67.4	87.4	62.5	72.9
ResNet-101 [11]	77.3	80.2	66.7	72.8	83.9	70.8	76.8	84.1	59.4	69.7	89.1	62.8	73.6
Multi-Evidence [5]	-	80.4	70.2	74.9	85.2	72.5	78.4	84.5	62.2	70.6	89.1	64.3	74.7
ML-GCN(ResNet50) [16]	*81.0*	***84.0***	*69.5*	*76.0*	***85.8***	*72.9*	*78.8*	*87.1*	*62.2*	*72.6*	***90.3***	*64.7*	*75.4*
Ours(ResNet50)	***81.4***	*83.6*	***70.5***	***76.5***	*84.0*	***74.5***	***79.0***	***88.4***	***64.3***	***73.2***	*89.5*	***65.6***	***75.7***
ML-GCN(ResNet101) [16]	82.3	83.3	**71.9**	77.2	84.7	**75.2**	79.7	87.3	**63.8**	73.7	90.1	**65.9**	76.1
Ours(ResNet101)	**82.9**	**86.8**	70.4	**77.7**	**87.6**	74.0	**80.2**	**90.1**	63.2	**74.3**	**91.3**	65.8	**76.5**

TABLE II
EXPERIMENTAL RESULTS ON PA-100K

Methods	All							Top-3					
	mAP	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***
ML-GCN(ResNet50) [16]	63.7	69.7	52.6	60.0	85.1	79.8	82.3	75.4	22.9	35.5	95.1	51.1	66.5
Ours(ResNet50)	**67.0**	**71.5**	**55.2**	**62.3**	**87.9**	**82.3**	**84.4**	**81.3**	**24.0**	**37.1**	**96.1**	**51.6**	**67.1**

TABLE III
ABLATION STUDIES ON MS-COCO(RESNET50)

Definition	All						
	mAP	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***
First Definition	81.0	**84.0**	69.5	76.0	**85.8**	72.9	78.78
Second Definition	**81.3**	81.4	**72.0**	**76.4**	83.2	**75.0**	**78.81**

TABLE IV
ABLATION STUDIES ON MS-COCO(RESNET101)

Definition	All						
	mAP	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***
First Definition	82.3	**83.3**	71.9	77.2	**84.7**	75.2	**79.7**
Second Definition	**82.6**	80.7	**74.2**	**77.3**	82.8	**76.0**	79.3

comprehensively and accurately. This paper proposes that using high-frequency category feature to supplement information for low-frequency category feature can improve the recognition performance of low-frequency categories. Next, this paper constructs a pair of asymmetric graphs using the two ways, and then uses a unified model to model asymmetric co-occurrence dependency on graph pair. Because of the strong feature extraction ability of GCN, model can learn classifiers which imply co-occurrence dependencies. These classifiers further improve the performance of multi-label recognition model. When defining asymmetric co-occurrence dependency graph pair, two threshold hyperparameters are used to increase the flexibility of our model. Both quantitative and qualitative results validated the advantages of our model.

TABLE V
ABLATION STUDIES ON PA-100K

Definition	All						
	mAP	***CP***	***CR***	***CF1***	***OP***	***OR***	***OF1***
First Definition	63.7	69.7	52.6	60.0	85.1	79.8	82.3
Second Definition	**66.3**	**70.4**	**56.0**	**62.4**	**86.4**	**80.3**	**83.2**

REFERENCES

[1] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the inception architecture for computer vision," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 2818-2826, doi: 10.1109/CVPR.2016.308.

[2] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[3] X. Wang, R. Girshick, A. Gupta and K. He, "Non-local neural networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 7794-7803, doi: 10.1109/CVPR.2018.00813.

[4] J. Hu, L. Shen and G. Sun, "Squeeze-and-excitation networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 7132-7141, doi: 10.1109/CVPR.2018.00745.

[5] W. W. Ge, S. Yang and Y. Yu, "Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 1277-1286, doi: 10.1109/CVPR.2018.00139.

[6] C. Lee, W. Fang, C. Yeh and Y. F. Wang, "Multi-label zero-Shot learning with structured knowledge graphs," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 1576-1585, doi: 10.1109/CVPR.2018.00170.

[7] S. Chen, Y. Chen, C. Yeh, and Y. F. Wang. "Order-free RNN with visual attention for multi-label classification," 2018 AAAI Conference on Artificial Intelligence, New Orleans, LA, 2018, pp. 6714–6721.

[8] Y. Li, C. Huang, C. C. Loy, and X. Tang. "Human attribute recognition by deep hierarchical contexts," 2016 European Conference on Computer Vision, Amsterdam, Netherlands, 2016, pp. 684–700, doi: 10.1007/978-3-319-46466-4_41.

[9] X. Yang, Y. Li, and J. Luo. “Pinterest board recommendation for twitter users,” 2015 Proceedings of the 2015 ACM Multimedia Conference, Brisbane, Australia, 2015, pp. 963–966, doi: 10.1145/2733373.2806375.

[10] K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition,” 2015 International Conference on Learning Representations, San Diego, CA, 2015, pp. 1–8.

[11] K. He, X. Zhang, S. Ren and J. Sun, “Deep residual learning for image recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[12] G. Huang, Z. Liu, L. Van Der Maaten and K. Q. Weinberger, “Densely connected convolutional networks,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 2261-2269, doi: 10.1109/CVPR.2017.243.

[13] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang and W. Xu, “CNN-RNN: a unified framework for multi-label image classification,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 2285-2294, doi: 10.1109/CVPR.2016.251.

[14] F. Zhu, H. Li, W. Ouyang, N. Yu and X. Wang, “Learning spatial regularization with image-level supervisions for multi-label image classification,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, pp. 2027-2036, doi: 10.1109/CVPR.2017.219.

[15] Z. Wang, T. Chen, G. Li, R. Xu and L. Lin, “Multi-label image recognition by recurrently discovering attentional regions,” 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 464-472, doi: 10.1109/ICCV.2017.58.

[16] Z. Chen, X. Wei, P. Wang and Y. Guo, “Multi-Label image recognition with graph convolutional networks,” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 5172-5181, doi: 10.1109/CVPR.2019.00532.

[17] T. N. Kipf and M. Welling. “Semi-supervised classification with graph convolutional networks,” 2017 International Conference on Learning Representations, Toulon, France, 2017, pp. 1–10.

[18] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. “Microsoft COCO: Common objects in context,” 2014 European Conference on Computer Vision, Zurich, Switzerland, 2014, pp. 740–755, doi: 10.1007/978-3-319-10602-1_48.

[19] X. Liu et al., “HydraPlus-net: Attentive deep features for pedestrian analysis,” 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 350-359, doi: 10.1109/ICCV.2017.46.

[20] J. Pennington, R. Socher, and C. Manning. “GloVe: Global vectors for word representation,” 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 2014, pp. 1532–1543, doi: 10.3115/v1/d14-1162.

[21] A. L. Maas, A. Y. Hannun, and A. Y. Ng. “Rectifier nonlinearities improve neural network acoustic models,” 2013 International Conference on Machine Learning, Atlanta, GA, 2013, pp. 1–6.

[22] Q. Li, Z. Han, and X. Wu. “Deeper insights into graph convolutional networks for semi-supervised learning,” 2018 AAAI Conference on Artificial Intelligence, New Orleans, LA, 2018, pp. 3538–3545.

[23] I. Loshchilov and F. Hutter. “SGDR: Stochastic gradient descent with warm restarts,” 2017 International Conference on Learning Representations, Toulon, France, 2017.

[24] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” in Neural Computation, vol. 9, no. 8, pp. 1735-1780, 15 Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[25] K. Cho, B. v. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio. “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, 2014, pp. 1724-1734.

[26] X. Zhang, X. Zhou, M. Lin and J. Sun, “ShuffleNet: An extremely efficient convolutional neural network for mobile devices,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 6848-6856, doi: 10.1109/CVPR.2018.00716.

[27] J. Zhang, Q. Wu, C. Shen, J. Zhang and J. Lu, “Multilabel image classification with regional latent semantic dependencies,” in IEEE Transactions on Multimedia, vol. 20, no. 10, pp. 2801-2813, Oct. 2018, doi: 10.1109/TMM.2018.2812605.

[28] H. Yang, J. T. Zhou, Y. Zhang, B. Gao, J. Wu and J. Cai, “Exploit bounding box annotations for multi-Label object recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 280-288, doi: 10.1109/CVPR.2016.37.

[29] Q. Li, M. Qiao, W. Bian and D. Tao, “Conditional graphical lasso for multi-label image classification,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, 2016, pp. 2977-2986, doi: 10.1109/CVPR.2016.325.

[30] X. Li, F. Zhao, and Y. Guo. “Multi-label image classification with a probabilistic label enhancement model,” 2014 Uncertainty in Artificial Intelligence, Quebec City, QC, Canada, 2014, pp. 430–439.