Contrastively Enforcing Distinctiveness for Multi-Label Classification

Motivation

The success of contrastive learning in single-label classifications motivates us to leverage this learning framework to enhance distinctiveness for better performance in multi-label image classification.

Gap

1. In single-label cases, an image usually contains one salient object, thus, the label of the object can also be viewed as the unique label of the image, so it is easy to define the positive or negative samples for an anchor image.

2. However, in multi-label cases, with a single image-level representation for an image, it is hard to define the positive or negative samples for an anchor image by its multiple labels.

Background

The Attention mechanism

$$\text{Att}(Q, K, V) = \omega(QK^T)V$$

where the dot product $(QK^T) \in \mathbb{R}^{n_q \times n_v}$ and $\omega(\cdot)$ is softmax function.

Multi-head attention is a extension of **Attention mechanism**, it allows the model to jointly attend to information from different representation subspaces at different positions

$$\text{MultiAtt}(Q, K, V) = \text{concat}(O_1, O_2, \cdots, O_h)W^o,$$
$$O_{i'} = \text{Att}(QW^q_{i'}, KW^k_{i'}, VW^v_{i'}) \text{ for } i' \in 1, \cdots, h,$$

Background

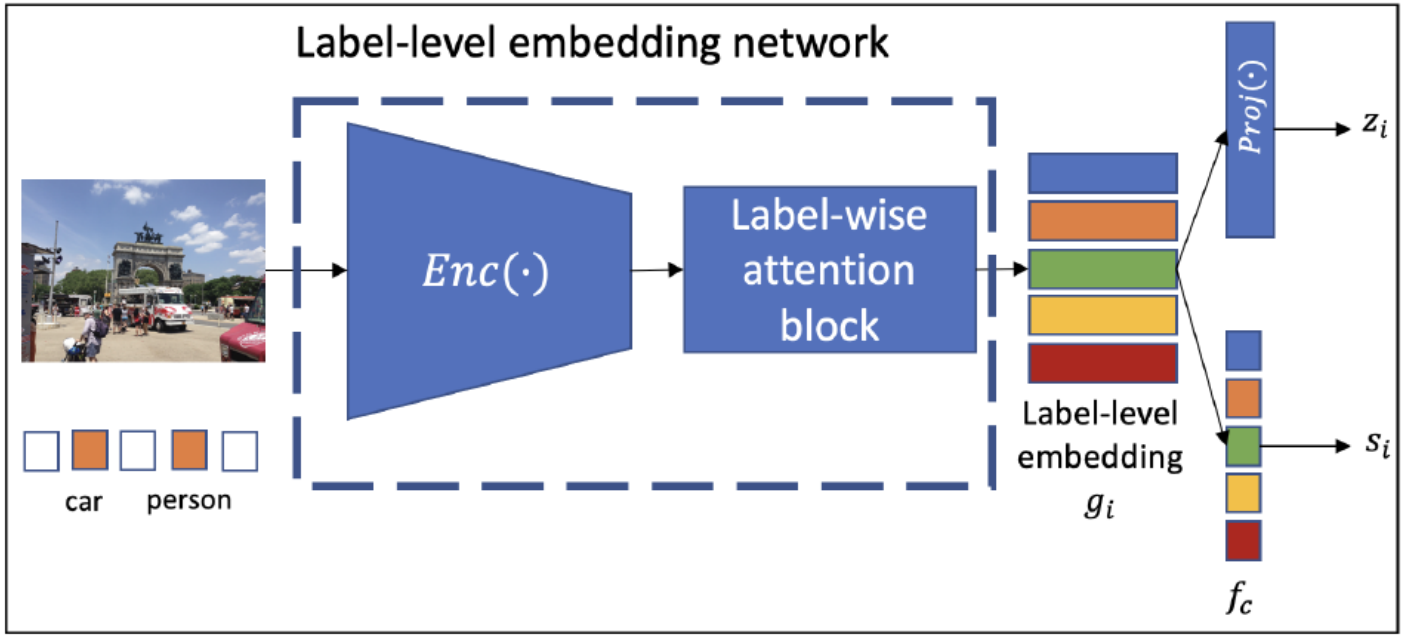
Following the architecture of the transformer, we define the following **multi-head attention block**:

$$\text{MultiAttBlock}(Q, K, V) = LayerNorm(Q' + Q'W^{q'}),$$
$$Q' = LayerNorm(\text{concat}(QW_1^q, \cdots, QW_h^q) + \text{MultiAtt}(Q, K, V))$$

Base on the multi-head attention block, we further define a **self-attention block** as follows:

$$\text{SA}(X) = \text{MultiAttBlock}(X, X, X)$$

Method



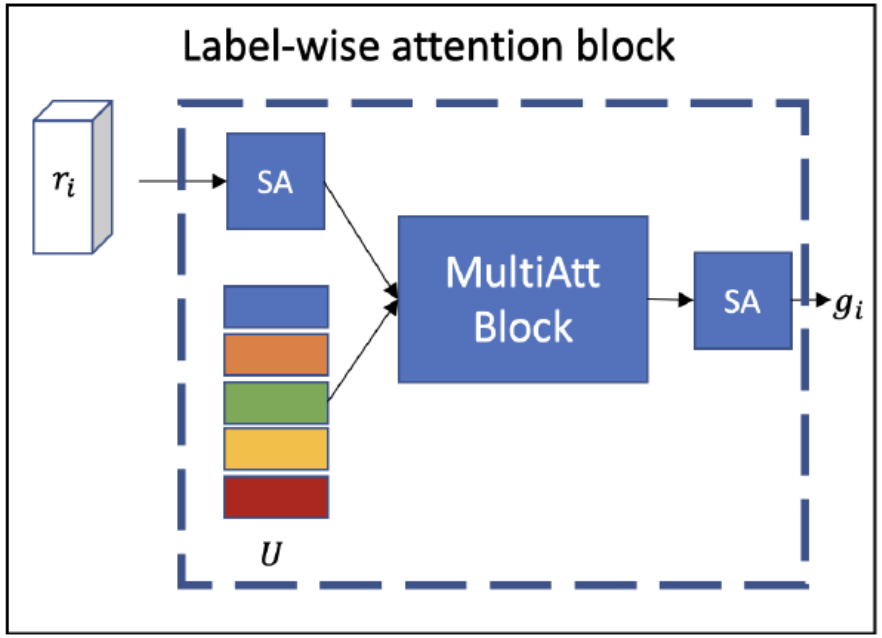


image-level embedding: $\boldsymbol{r}_i \in \mathbb{R}^{WH \times C}$

global label embeddings: $\boldsymbol{U} \in \mathbb{R}^{L \times C}$

label-level representations: $\boldsymbol{g}_i \in \mathbb{R}^{L \times D}$

SA: Self-Attention block

$$r_i = \mathrm{SA}(r_i); \quad g_i = \mathrm{MultiAttBlock}(U, r_i, r_i); \quad g_i = \mathrm{SA}(g_i).$$

Method

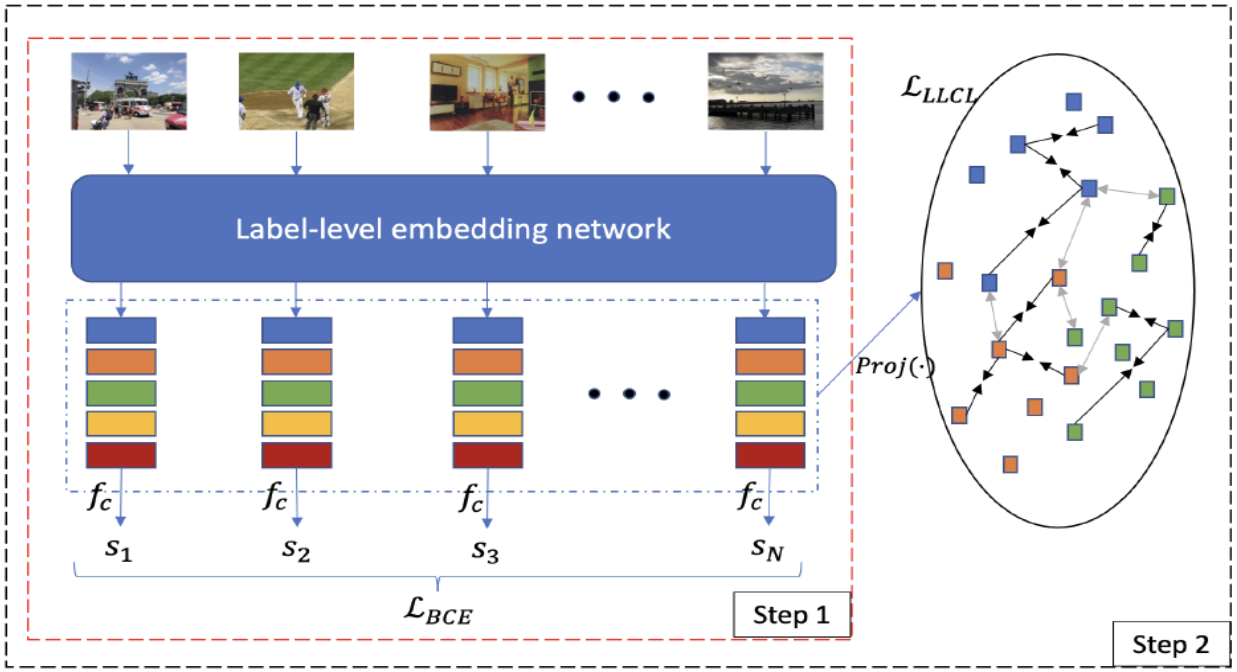


Figure 2: **MulCon** has two steps during training: pretraining and contrastive finetuning. The first step is to train the label-level embedding network with binary cross-entropy loss ($\mathcal{L}_{BCE}$) to effectively decompose an input image into several semantic components so that the first component corresponds to the first label, etc. The second step is to finetune the previously trained network with contrastive loss ($\mathcal{L}_{LLCL}$) and $\mathcal{L}_{BCE}$ to improve the quality of label-level embedding.

Method

Classification Loss

$$\mathcal{L}_{BCE} = \sum_{j=1}^{L} y_{ij} \log s_{ij} + (1 - y_{ij}) \log(1 - s_{ij})$$

Label-level Contrastive Loss

aggregating the label-level embeddings of all the images into set: $Z = \{z_{ij} \in \mathbb{R}^{d_z} \mid i \in \{1, \cdots, N\}; j \in \{1, \cdots, L\}\}$

the set of the ground-truth labels: $Y = \{y_{ij} \in \{0, 1\} \mid i \in \{1, \cdots, N\}; j \in \{1, \cdots, L\}\}$

$$I = \{z_{ij} \in Z \mid y_{ij} = 1\}$$

$$A(i, j) = I \backslash z_{ij}$$

$$P(i, j) = \{z_{kj} \in A(i, j) \mid y_{kj} = y_{ij} = 1\}$$

$$\mathcal{L}_{LLCL}^{ij} = \frac{-1}{|P(i,j)|} \sum_{z_p \in P(i,j)} \log \frac{\exp(z_{ij} \cdot z_p / \tau)}{\sum_{z_a \in A(i,j)} \exp(z_{ij} \cdot z_a / \tau))},$$

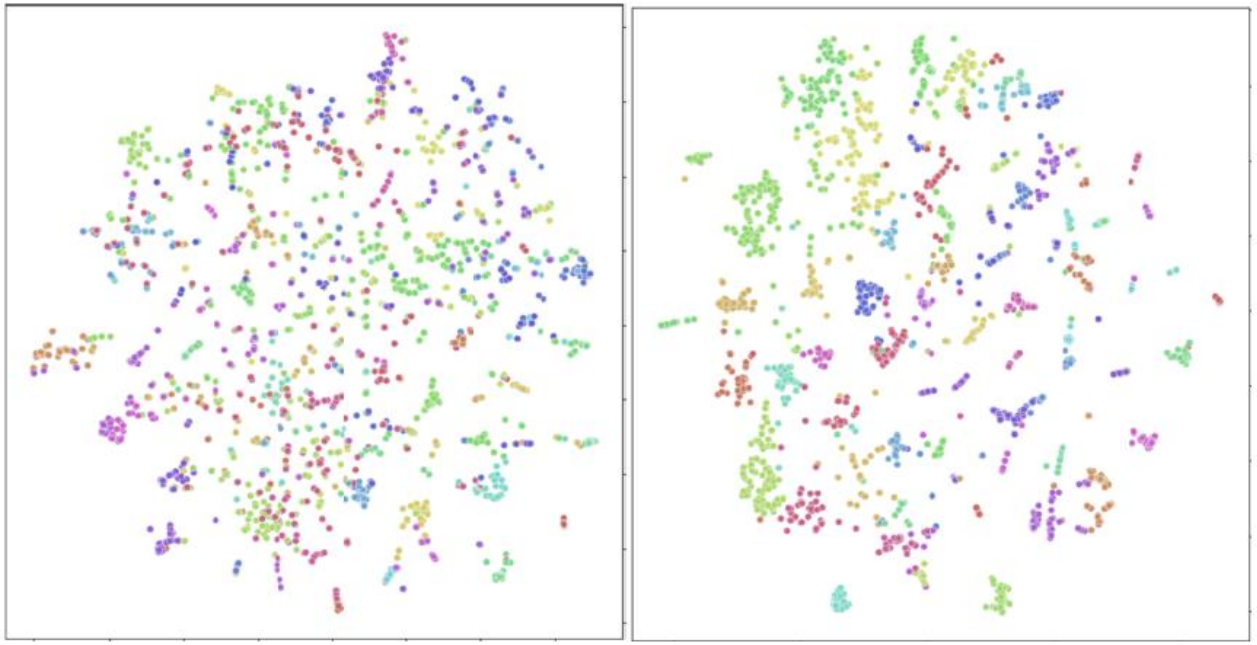
$$\mathcal{L}_{LLCL} = \sum_{z_{ij} \in I} \mathcal{L}_{LLCL}^{ij}.$$

$$\mathcal{L} = \mathcal{L}_{BCE} + \gamma \mathcal{L}_{LLCL}.$$

Why and How does Contrastive Loss Help?

BCE 分类损失常用于多标签分类问题，对于每个标签，可以被视为使用特定分类器独立分类，它让每个分类器都只关注特定标签的分类，而不关心不同标签的判别性特征，如果能在分类过程中考虑这一点，就可能提升它的性能，在 label-level 的 representation 间加上对比学习就是出于这一考虑.

Visualization for image components trained with only BCE loss (left) and with the combination of BCE loss and contrastive loss (right)



Method

However, being over distinct in the embedding space is not always a good thing in multi-label classification !

a simple training strategy

Two Step:

1. In the pre-training step (Step 1), we pretrain the backbone and label-level embedding network and the classification network of MulCon with the BCE loss only.
2. In the contrastive learning step (Step 2), we then plug in the contrastive projection network with LLCL but also keep the BCE loss.

In the first step, the BCE loss learns the label-level embeddings freely and implicitly obtains semantic structure by the effect of several attention blocks. After the embeddings are learned, we finetune them with LLCL to enforce distinctiveness of the embeddings.

Experiment

Method	Resolution	All							Top-3					
		mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
Multi Evidence	448 × 448	-	80.4	70.2	74.9	85.2	72.5	78.4	84.5	62.2	70.6	89.1	64.3	74.7
CADM	448 × 448	82.3	82.5	72.2	77.0	84.0	75.6	79.6	87.1	63.6	73.5	89.4	66.0	76.0
ML-GCN	448 × 448	83.0	**85.1**	72.0	78.0	85.8	75.4	80.3	**89.2**	64.1	74.6	90.5	66.5	76.7
KSSNet	448 × 448	83.7	84.6	73.2	77.2	87.8	76.2	81.5	-	-	-	-	-	-
MS-CMA	448 × 448	83.8	82.9	74.4	78.4	84.4	77.9	81.0	86.7	64.9	74.3	90.9	67.2	77.2
MCAR	448 × 448	83.8	85.0	72.1	78.0	**88.0**	73.9	80.3	88.1	65.5	75.1	**91.0**	66.3	76.7
MulCon (Ours)	448 × 448	**84.9**	84.0	**74.8**	**79.2**	85.6	**78.0**	**81.6**	87.8	**65.9**	**75.3**	90.5	**67.9**	**77.6**
SSGRL	576 × 576	83.8	**89.9**	68.5	76.8	**91.3**	70.8	79.7	**91.9**	62.5	72.7	**93.8**	64.1	76.2
C-Trans	576 × 576	85.1	86.3	74.3	79.9	87.7	76.5	81.7	90.1	65.7	76.0	92.1	**71.4**	77.6
ADD-GCN	576 × 576	85.2	84.7	75.9	80.1	84.9	79.4	82.0	88.8	66.2	75.8	90.3	68.5	77.9
MulCon (Ours)	576 × 576	**86.3**	84.7	**77.3**	**80.8**	85.9	**79.9**	**82.8**	88.6	**67.2**	**76.5**	91.0	68.8	**78.4**

Table 1: Results on the COCO dataset. The best scores are highlighted in boldface. More important metrics including mAP, CF1, and OF1 are highlighted in grey.

Experiment

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
RDAL	98.6	97.4	96.3	96.2	75.2	92.4	96.5	97.1	76.5	92.0	87.7	96.8	97.5	93.8	98.5	81.6	93.7	82.8	98.6	89.3	91.9
RARL	98.6	97.1	97.1	95.5	75.6	92.8	96.8	97.3	78.3	92.2	87.6	96.9	96.5	93.6	98.5	81.6	93.1	83.2	98.5	89.3	92.0
MCAR	99.7	**99.0**	98.5	98.2	85.4	96.9	97.4	**98.9**	83.7	95.5	88.8	99.1	98.2	95.1	99.1	84.8	97.1	**87.8**	98.3	94.8	94.8
SSGRL	99.7	98.4	98.0	97.6	85.7	96.2	98.2	98.8	82.0	98.1	89.7	98.8	98.7	97.0	99.0	86.9	98.1	85.8	99.0	93.7	95.0
ASL	**99.9**	98.4	98.9	**98.7**	**86.8**	98.2	**98.7**	98.5	83.1	98.3	89.5	98.8	**99.2**	**98.6**	**99.3**	**89.5**	99.4	86.8	99.6	95.2	**95.8**
MulCon	99.8	98.3	**99.3**	98.6	83.3	**98.4**	98.0	98.3	**85.8**	**98.3**	**90.5**	**99.3**	98.9	96.6	98.8	86.3	**99.8**	87.3	**99.8**	**96.1**	95.6

Table 4: Results on VOC07. Best results are highlighted in boldface.

Experiment

Method	mAP	CF1	OF1
FitsNet	57.4	54.9	70.4
attention-transfer	57.6	55.2	70.3
s-CLs	60.1	58.7	73.3
MS-CMA	61.4	60.5	73.8
SRN	62.0	58.5	73.4
MulCon (Ours)	**63.9**	**61.8**	**74.8**

Table 2: Results on NUS-WIDE dataset. The best scores are highlighted in boldface.

Method	mAP	CF1	OF1
R101 + BCE	80.8	76.2	79.2
R101 + BCE + SCL	80.8	76.0	79.1
LLEN + BCE	83.8	78.8	81.1
LLEN + BCE + LLCL	83.7	78.8	81.1
MulCon	84.9	79.2	81.6

Table 3: Ablation study of different variants and training policies of MulCon.

R101 + BCE: The backbone model, Resnet101, trained with the BCE loss
R101 + BCE + SCL: Resnet101 trained with the BCE and supervised-CL losses
LLEN + BCE: The label-level embedding network (LLEN) with R101 as the backbone trained with BCE
MulCon: The complete model of MulCon trained with the two-step policy
LLEN + BCE + LLCL: Same as MulCon except that two-step policy is not used

Experiment

Figure 4: Top-4 related images retrieved given an query image and label on COCO dataset. The results for our full model (MulCon) are on the left, and the results for our model without contrastive loss (MulCon with BCE only) are on the right. The label under each retrieved image is the one corresponding to the embedding closest to the picked query embedding.

Experiment

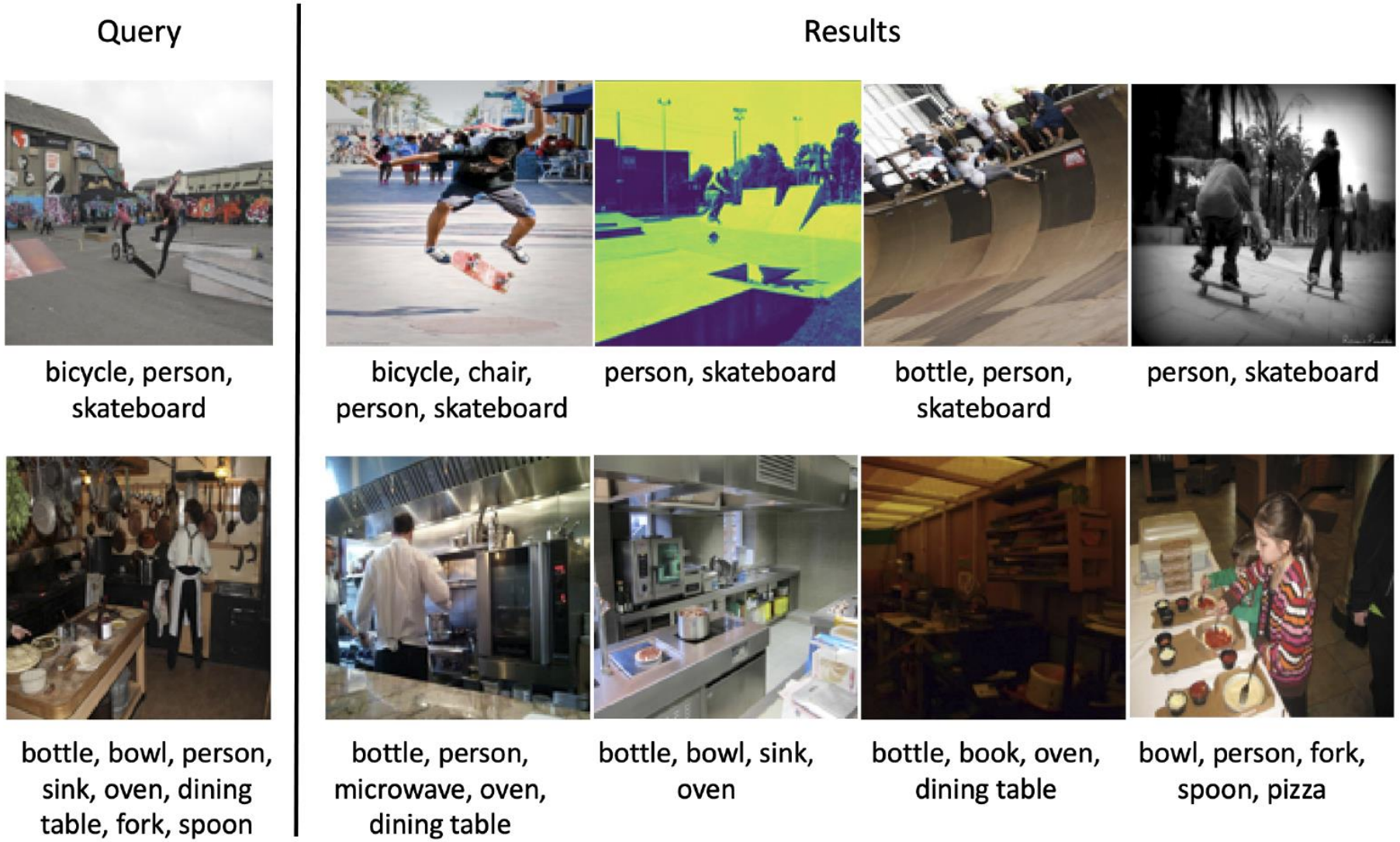


Figure 5: Top-4 related images retrieved given an query image and multiple labels on COCO dataset.

Thanks