

MMfruit - OpenImage 2019 1st solution

Yu Liu, Guanglu Song, Yuhang Zang, Yan Gao, Junjie Yan, Chen Change Loy, Xiaogang Wang

Multimedia Laboratory, The Chinese University of Hong Kong
Multimedia Laboratory, Nanyang Technological University
Sensetime X-Lab



Team members

MMfruit Team



Yu Liu



Guanglu Song



Yuhang Zang



Yan Gao



Junjie Yan

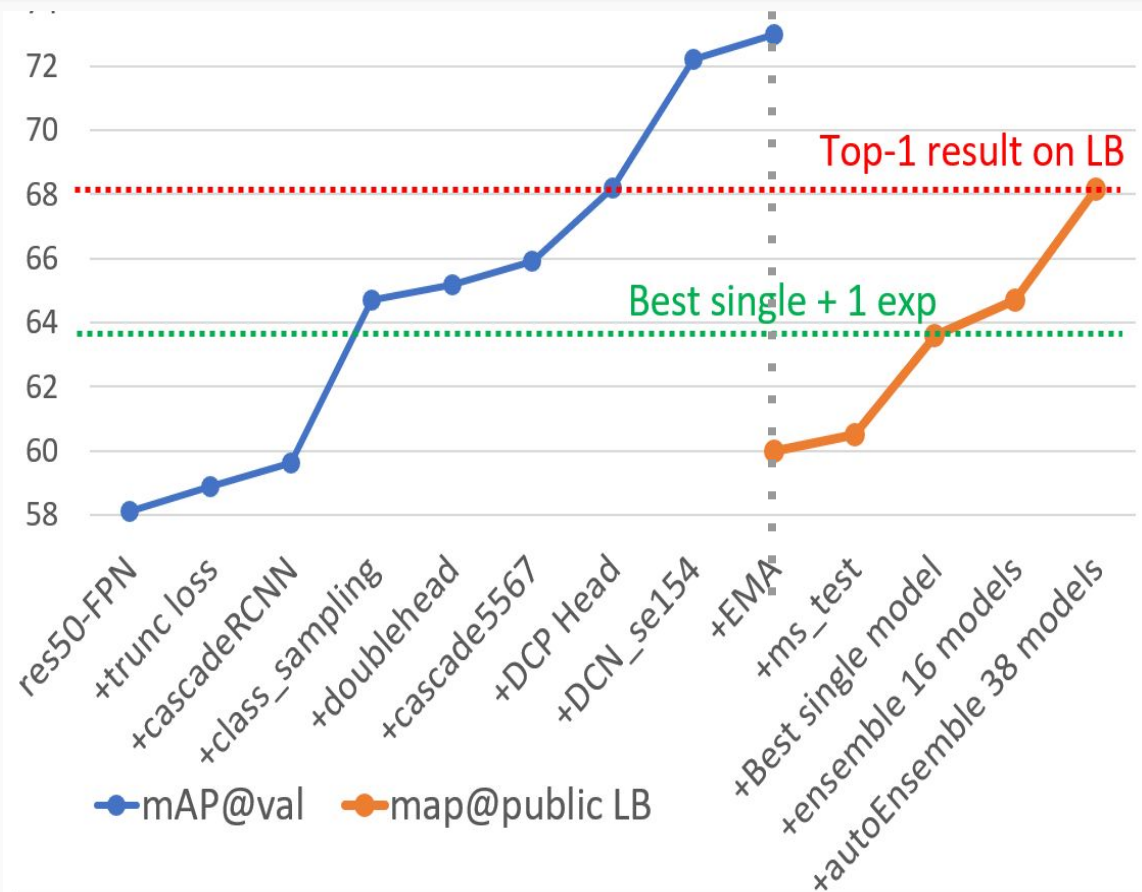


Chen Change Loy



Xiaogang Wang

Results Breakdown



- 39 models in total (including exps):
- Data distribution:
 - 26 trained by all classes
 - 3 expert models (low AP)
 - 2 models from COCO
 - 8 models from O365
- Framework:
 - 27 from pytorch
 - 10 from tensorflow
- 32~512 accelerators for each model
- 2 images on each accelerator

Solutions

- Multiple Large Backbones
- **Gradient Decoupling**
- **Class Sampling & Full Batch**
- Augmentation (Segmentation Label)
- Truncated Loss
- Multiscale Testing
- **Adj-soft NMS**
- **Expert Model**
- Weakly & Fully Supervised Pipeline
- **Auto Ensemble**

Multiple Large Backbones

Model Zoo

ResNet family

ResNet50

ResNet152

SEResNet family

SEResNext101

DCN-SEResNext101

DCN-SEResNet154

NAS family

NASNet

NAS-FPN

ResNext family

ResNext101

ResNext152

DCN-ResNext101

DCN-ResNext152

EfficientNet family

EfficientNetB7

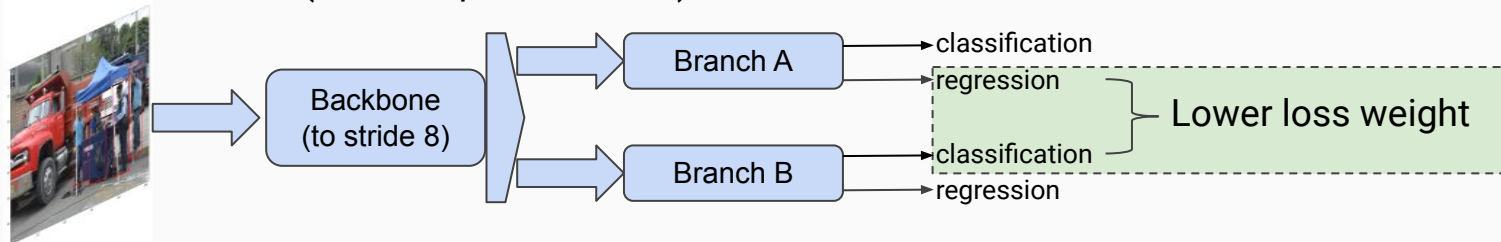
WideEfficientNetB7

Expert Model family

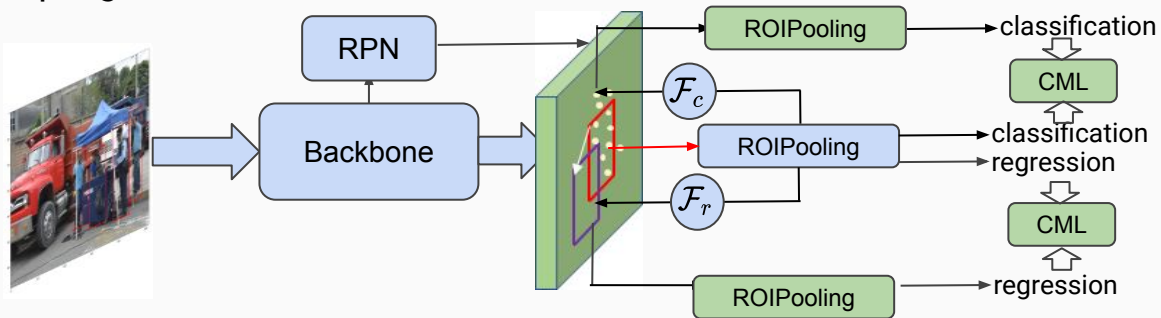
SEResNet154

Gradient Decoupling

Decoupling Backbone (Naive implementation)



Decoupling Head



Gradient Decoupling

Learn the offset for classification and regression separately.

$$C = \mathcal{F}_c(F; \theta_c) \quad C \in \mathbb{R}^{k \times k \times 2}$$

$$R = \mathcal{F}_r(F; \theta_r) \quad R \in \mathbb{R}^{1 \times 1 \times 2}$$

CML for classification: $L_c = |S_o - S + m_c|_+$

for regression: $L_r = |IoU_o - IoU + m_r|_+$

Exp on OpenImage2019

Model	DCP	Val	Public LB
ResNet50		64.64	49.79
ResNet50	√	68.18	52.55
DCN-ResNext101		68.70	55.05
DCN-ResNext101	√	71.71	58.59
DCN-SENet154		71.13	57.77
DCN-SENet154	√	72.19	60.5

Exp on COCO 2017 (FPN)

Model	A1	A2	A3	CML	DCP	mAP (IOU=0.50:0.95)
ResNet50						36.1
ResNet50	√					37.3
ResNet50		√				38.0
ResNet50			√			38.5
ResNet50					√	39.7
ResNet50				√	√	40.8

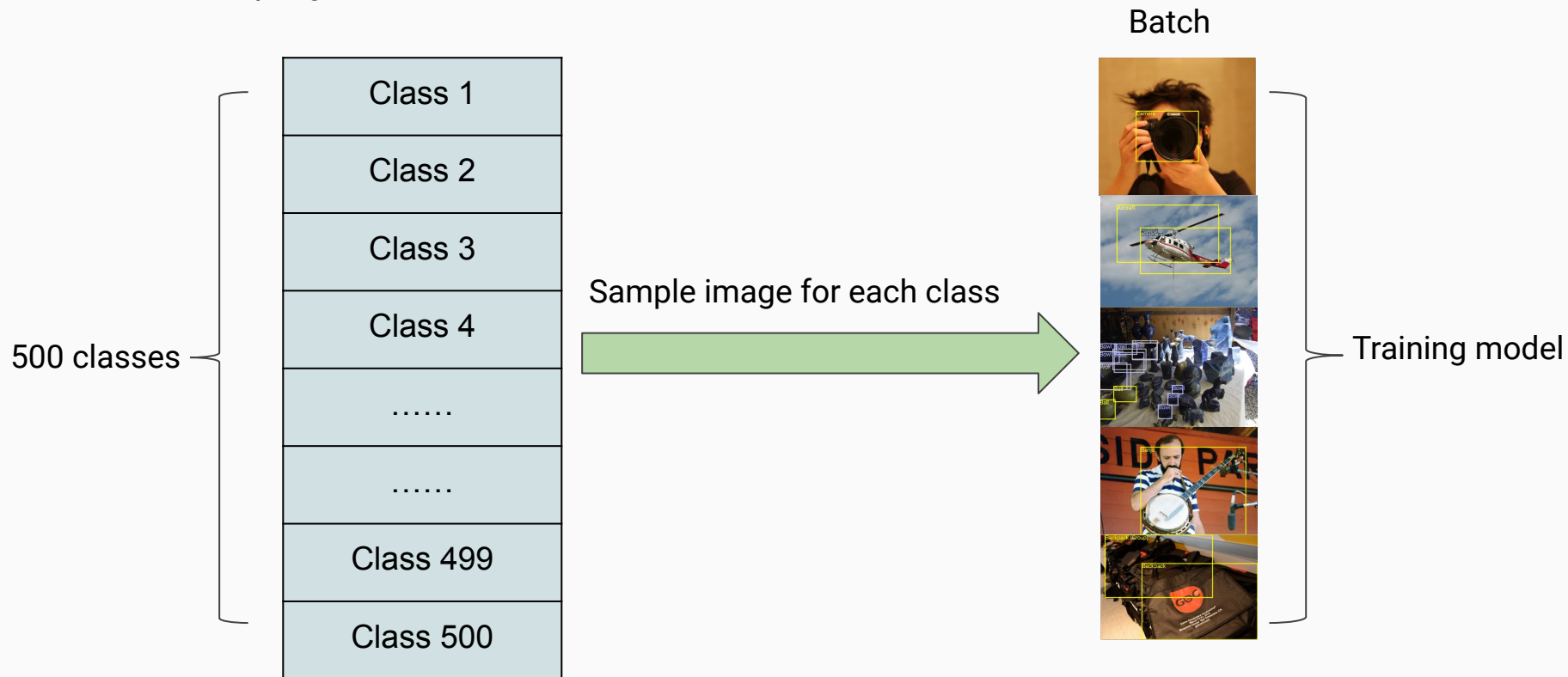
A1: separate classification and regression

A2: Deformable ROI Pooling for classification and ROI Align Pooling for regression.

A3: Deformable ROI Pooling for classification and Deformable ROI Pooling for regression.

Class Sampling & Full Batch

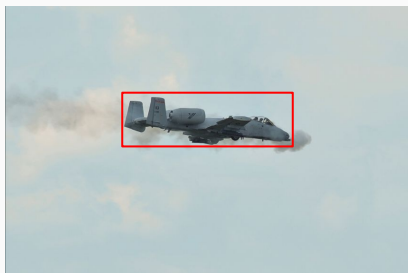
Class-aware sampling & full batch:



Augmentation (Segmentation Label)

Elaborate Augmentation

Select an image and a bounding box



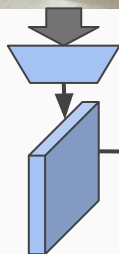
Random
rotation



Sample a
specific
scale
Crop



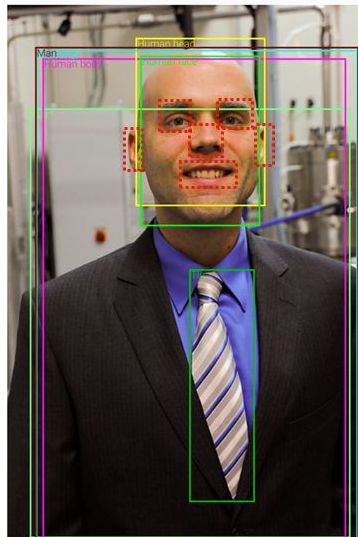
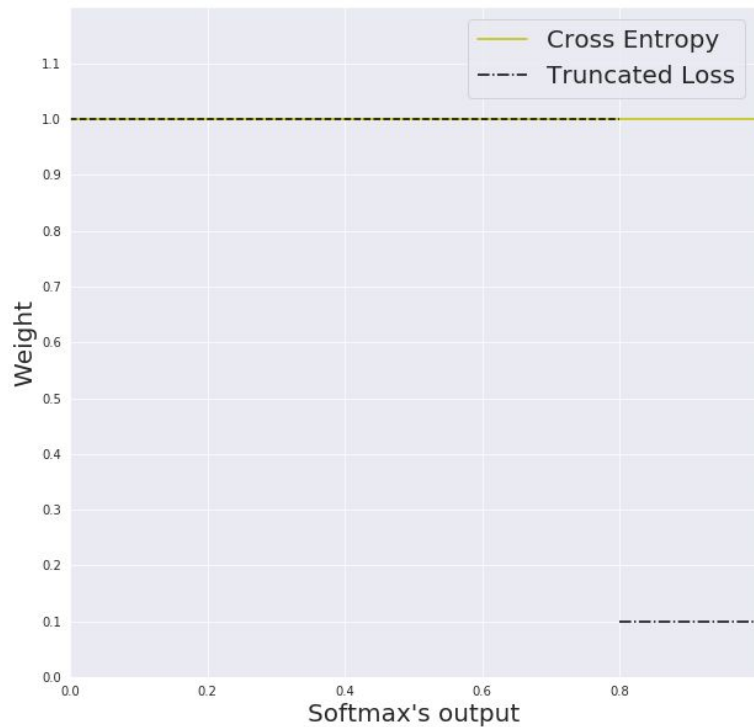
Copy-and-Paste Augmentation



Augmentation



Truncated Loss



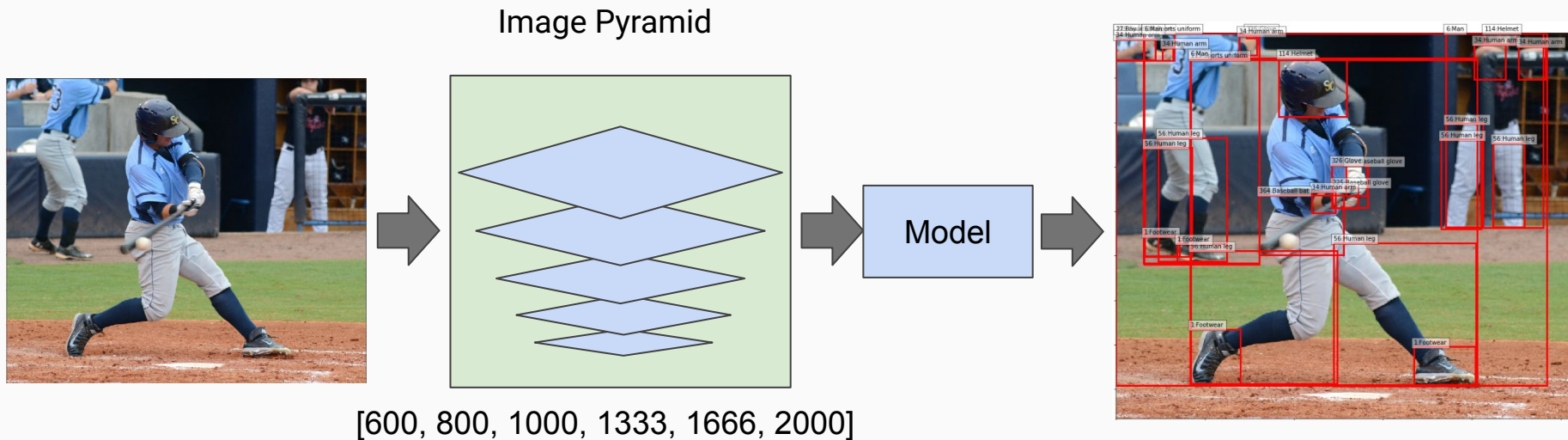
Missing (red dashed box):
human eye, ear, nose, mouth,

...

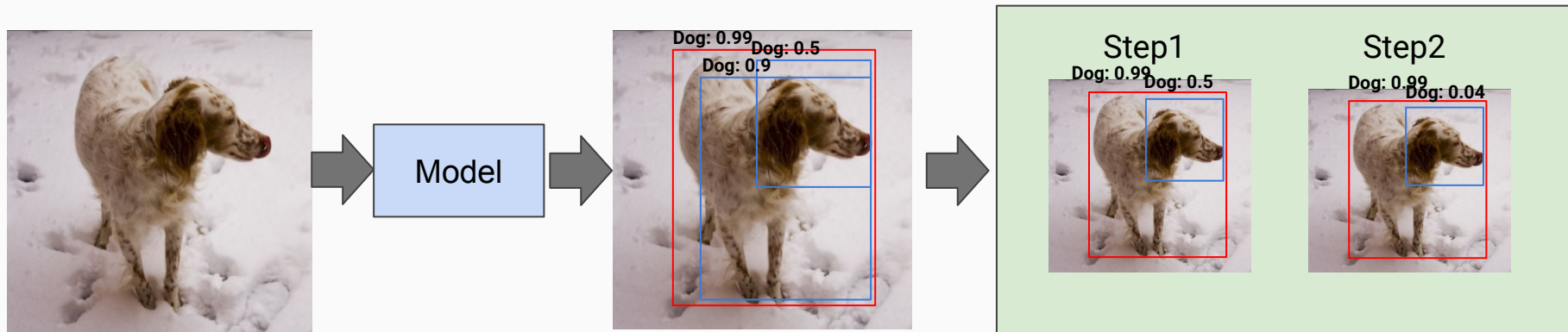


Missing (red dashed box):
wheel, tire, tree, light, ...

Multiscale Testing



Adj-soft NMS



Input : $\mathcal{B} = \{b_1, \dots, b_N\}, \mathcal{S} = \{s_1, \dots, s_N\}, T = 0.5$

Begin

$\mathcal{D} \leftarrow \{\}$

$\mathcal{D} \leftarrow \text{Step1}(\mathcal{B}, \mathcal{S}, \mathcal{D}, T)$ NMS

$\mathcal{F} \leftarrow \{\}$

$\mathcal{F} \leftarrow \text{Step2}(\mathcal{B}, \mathcal{S}, \mathcal{D})$ Soft-NMS

return \mathcal{F}, \mathcal{S}

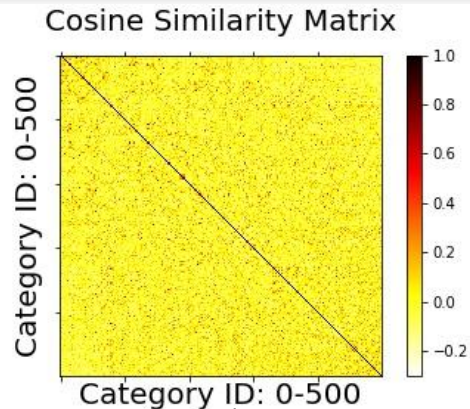
end

Model	Adj-soft NMS	Public LB
{4models}		59.40
{4models}	✓	60.35

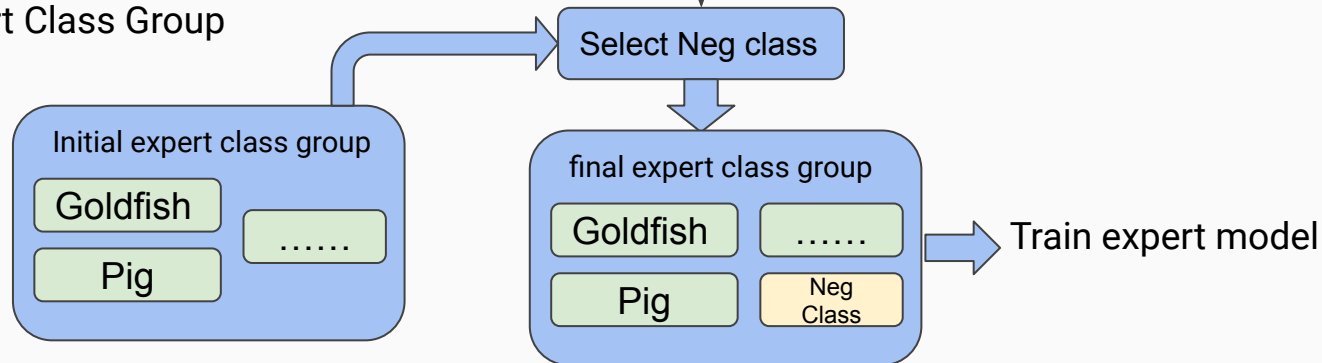
Expert Model

(1) Build Similarity Matrix

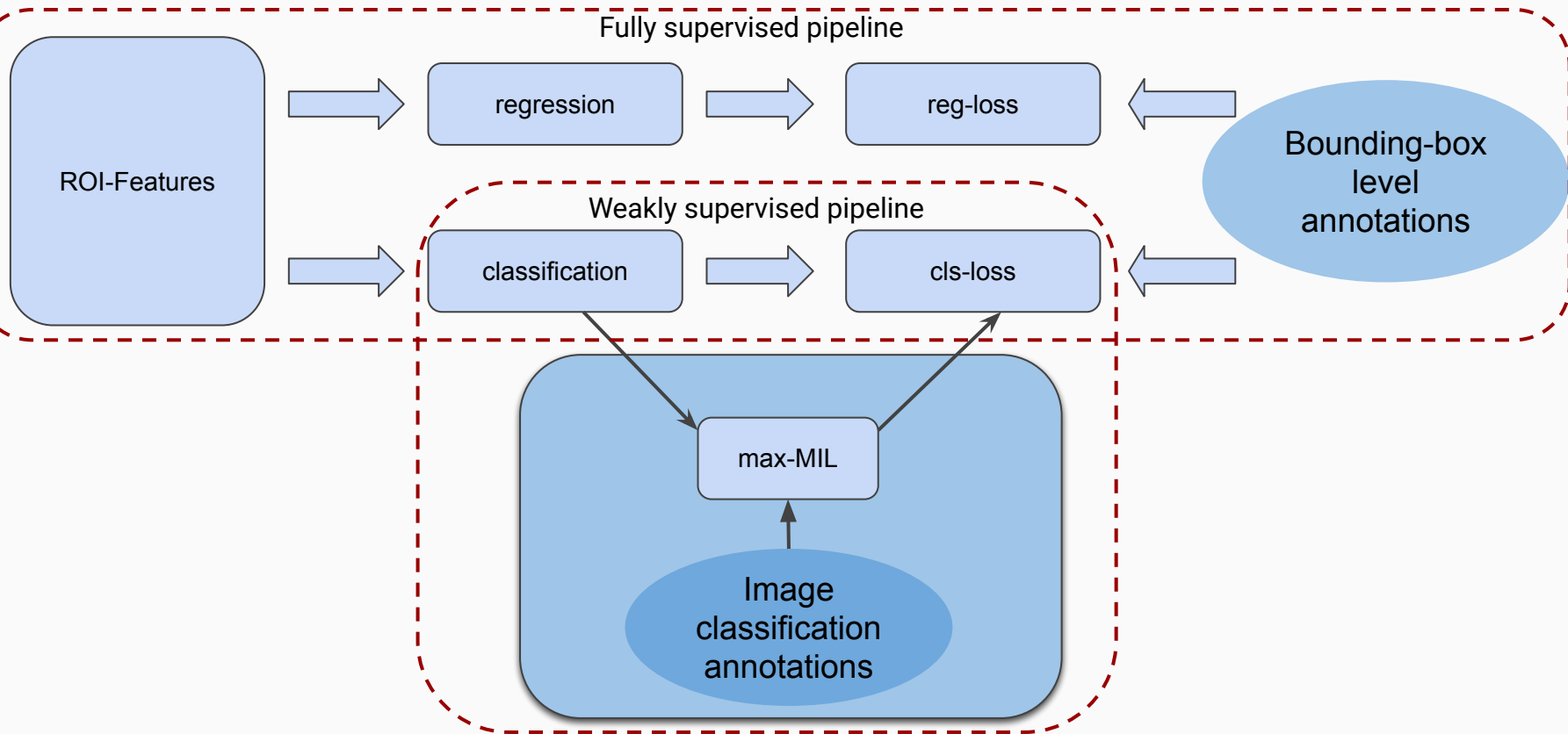
Class centroid matrix



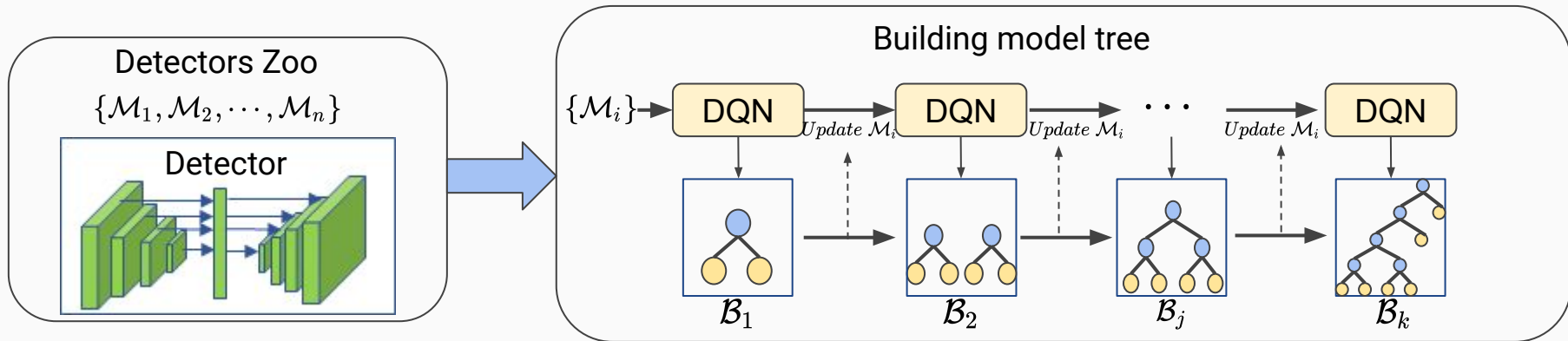
(2) Generate Expert Class Group



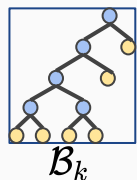
Weakly & Fully Supervised Pipeline



Auto Ensemble



●: Functions set, such as NMS, Adj-soft NMS, and so on



➔ Model inference from leaf to root

Selecting the operation in each ● via greedy algorithm

- Confusing definitions:



- For more please read our solutions

Thanks, Q & A