DataLab Cup 2: Object Detection

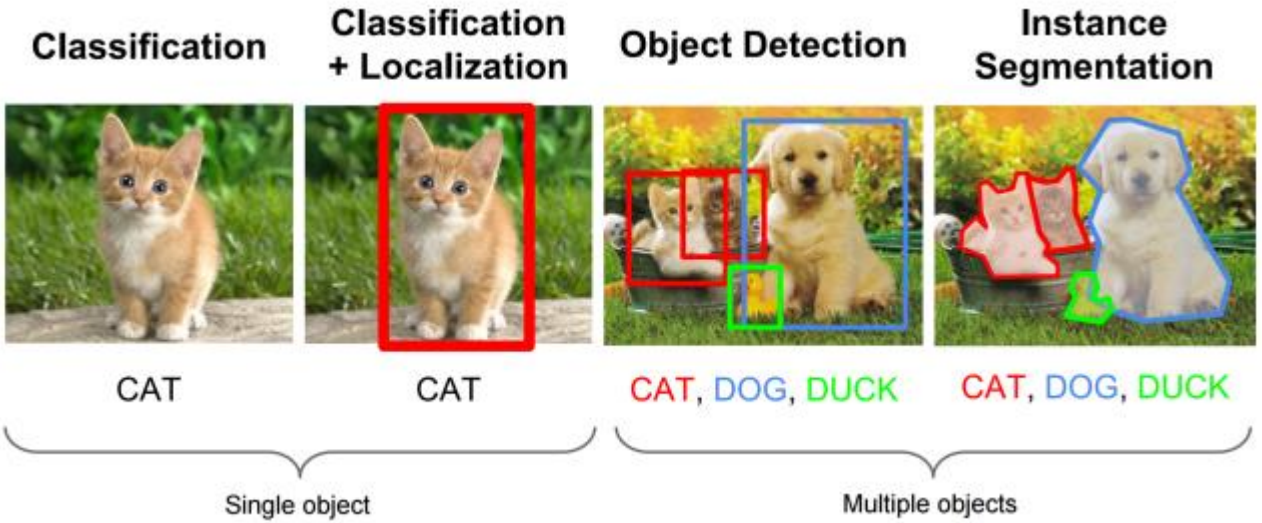
Datalab

Outline

- Competition Information
- Evaluation metric
  - Mean Average Precision (mAP)
- Hints
- Precautions
- Competition Timeline

Competition Information

- Object Detection
  - In this competition, we are going to train an object detection model to detect objects in an image.



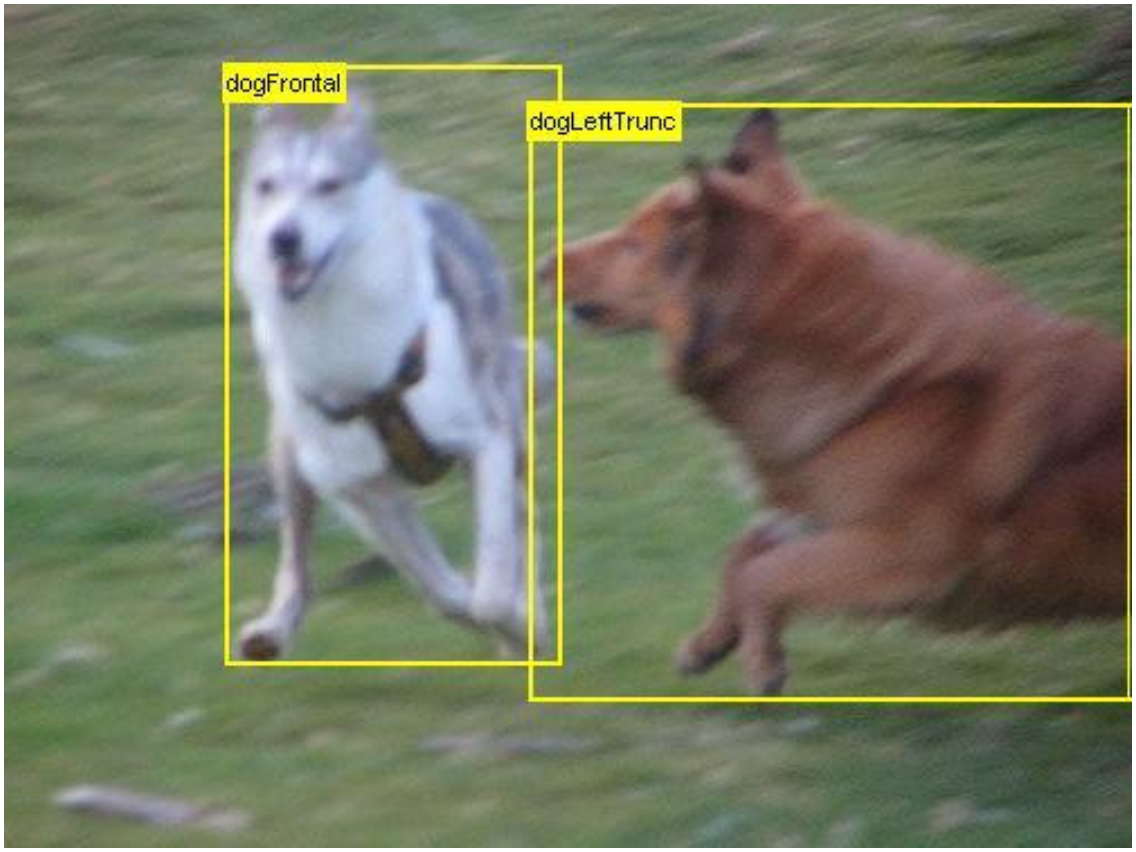
Competition Information

- Dataset
  - PASCAL VOC 2007

The PASCAL Visual Object Classes Challenge 2007

Competition Information

- Dataset
  - PASCAL VOC 2007



Competition Information

- Dataset
  - PASCAL VOC 2007
    - Train/Val data: 5011
      - Each row contains one image and its bounding boxes.
      - filename, $(x_{min}, y_{min}, x_{max}, y_{max}, \text{label}) * \text{object\_num}$

```
000012.jpg 156 97 351 270 6
000016.jpg 92 72 305 473 1
000017.jpg 185 62 279 199 14 90 78 403 336 12
000019.jpg 231 88 483 256 7 11 113 266 259 7
```

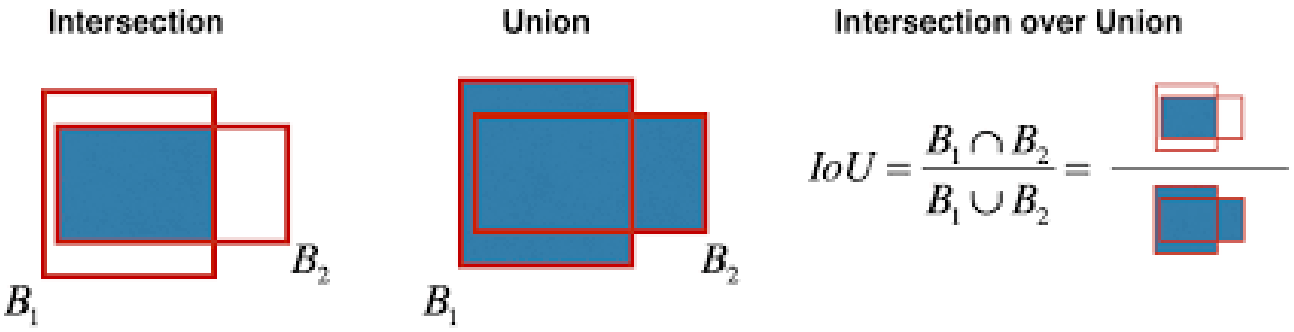
    - Test data: 4952
      - filename

```
000001.jpg
000002.jpg
000003.jpg
```

Evaluation Metric

Mean Average Precision (mAP)

- Intersection over Union (IoU)
  - A metric to evaluate the effectiveness of predict bounding box comparing to the ground truth.



Evaluation Metric

Mean Average Precision (mAP)

- Confusion matrix reminder
  - **True positive (TP)**: A correct detection. Detection with $\text{IoU} \geq threshold.$
  - **False positive (FP)** : A wrong detection. Detection with $\text{IoU} < threshold.$
  - **False Negative (FN)**: A ground truth not detected.
  - **True Negative (TN)**: A correct misdetection. Does not apply in evaluation.

Evaluation Metric

Mean Average Precision (mAP)

- Precision x Recall curve
  - Precision: the percentage of correct positive predictions.

$$Precision = \frac{TP}{TP + FP} = \frac{TP}{all\ detections}$$

  - Recall: the percentage of true positive detected among all ground truths.

$$Recall = \frac{TP}{TP + FN} = \frac{TP}{all\ ground\ truths}$$

Evaluation Metric

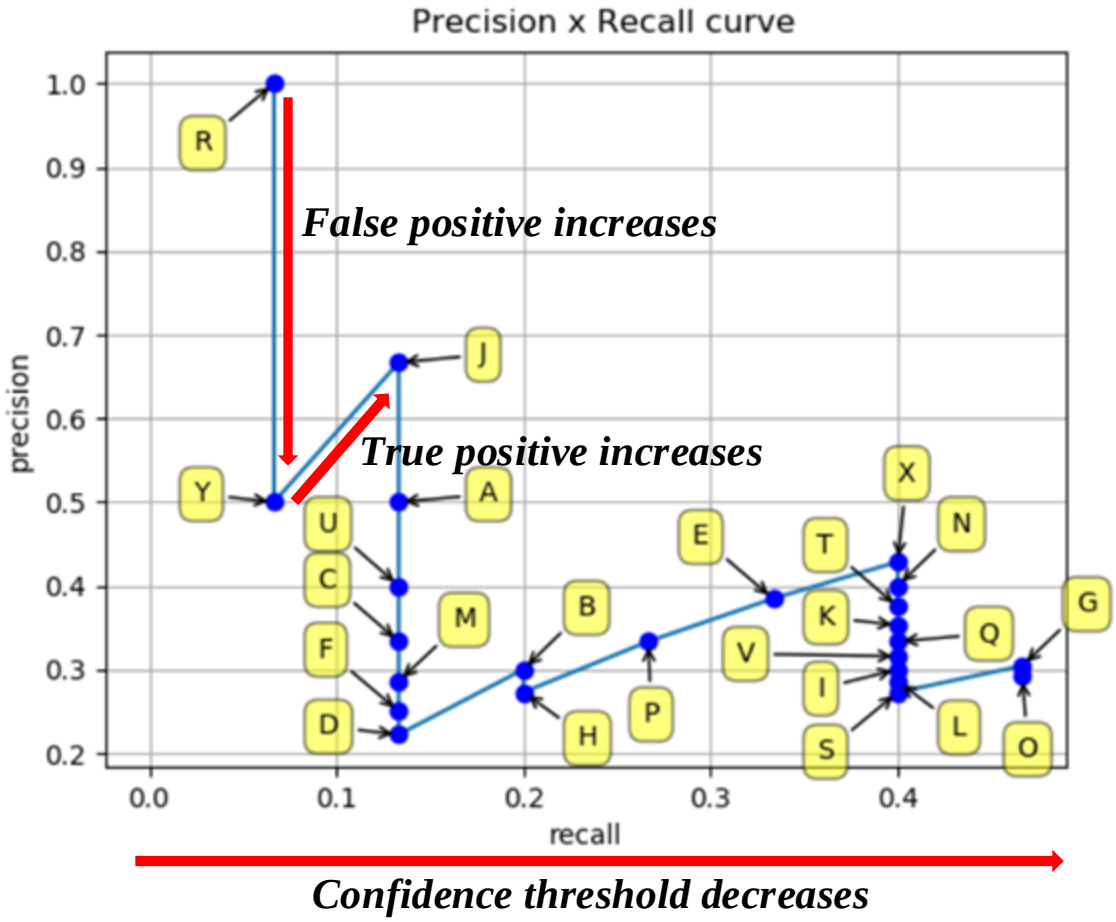
Mean Average Precision (mAP)

- Precision x Recall curve
  - An object detector of a particular class is considered good if its precision stays high as recall increases.
  - It means that if you vary the confidence threshold, the precision and recall will still be high.

Evaluation Metric

Mean Average Precision (mAP)

- Precision x Recall curve



Evaluation Metric

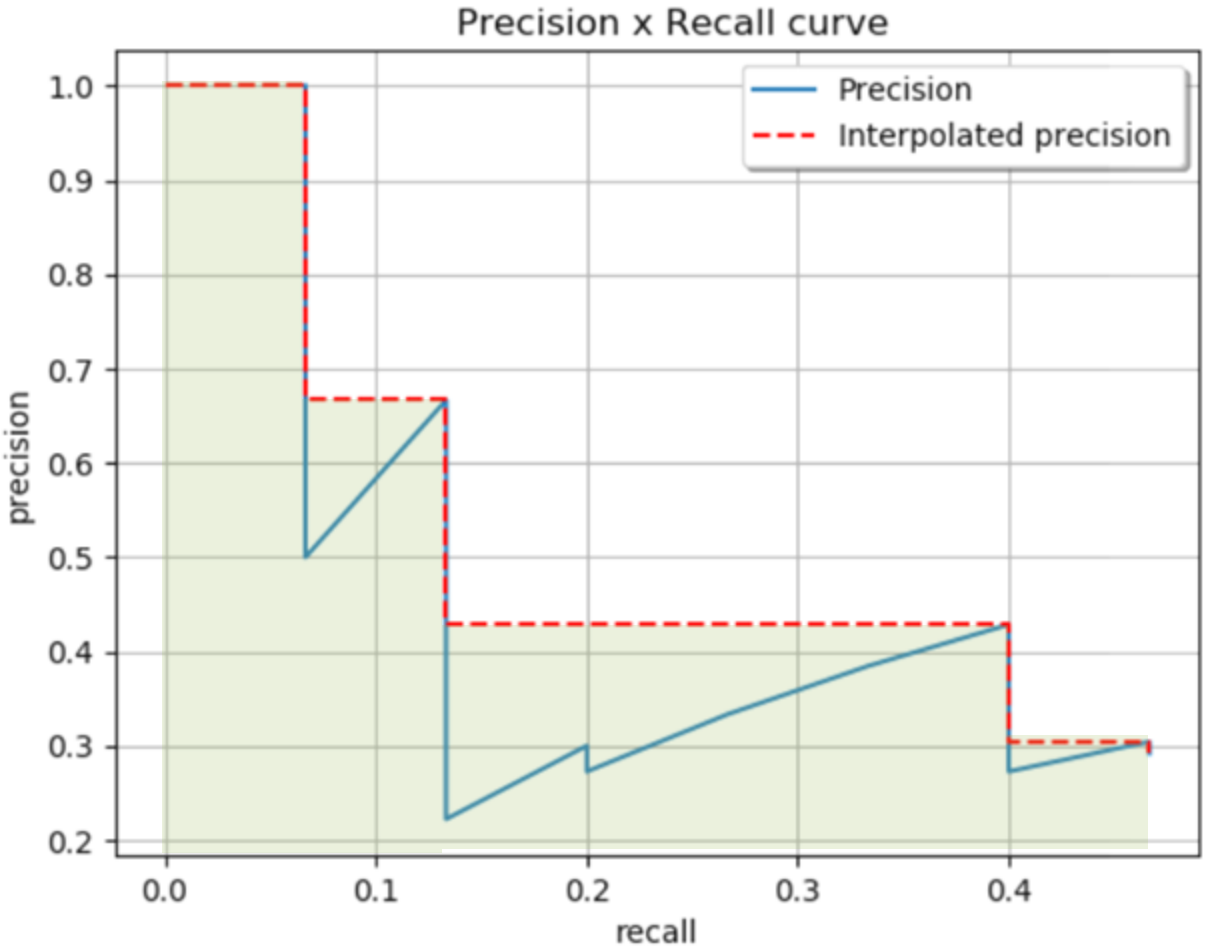
Mean Average Precision (mAP)

- Average Precision (AP)
  - Smooth the Precision-recall curve and calculate the area under curve (AUC).

Evaluation Metric

Mean Average Precision (mAP)

- Average Precision (AP)



Evaluation Metric

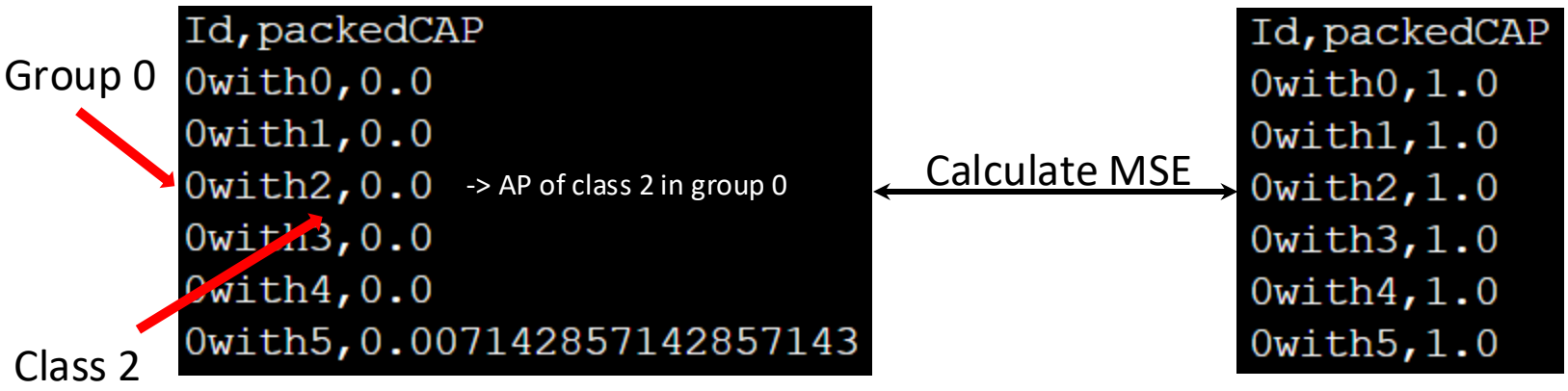
Mean Average Precision (mAP)

- Mean Average Precision (mAP)
  - Calculate the Average Precision for every class and average them.

Evaluation Metric

Mean Average Precision (mAP)

- Mean Average Precision (mAP)
  - In this competition, we divide testing data into 10 groups and calculate the mAP of all classes.
  - After deriving the mAP of each class in 10 groups, we compare the result with ground truth and use the mean square error as the final score.



Evaluation Metric

Mean Average Precision (mAP)

- Mean Average Precision (mAP)
  - For more detailed explanation of mAP, please see https://github.com/rafaelpadilla/Object-Detection-Metrics

Hints

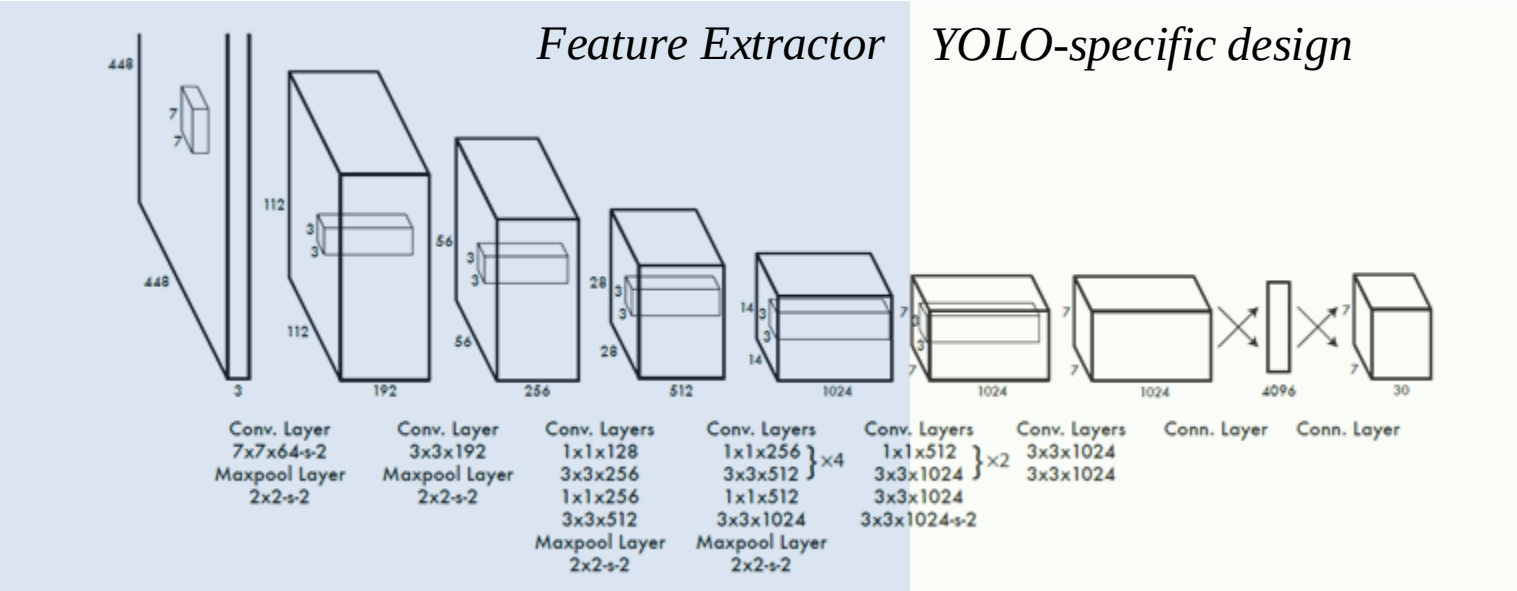
1. Transfer learning
2. Data augmentation
3. Training strategy
4. Other object detection models

Hints

1. Transfer learning
   - Training from scratch is nearly impossible for object detection
   - How to load pre-trained model is already described in lab: style transfer
   - You can see all the pre-trained models provided by Keras here: https://www.tensorflow.org/api_docs/python/tf/keras/applications

Hints

1. Transfer learning
   - Feel free to replace the feature extractor with other pre-trained model
   - Be careful that different models require different data preprocess



Hints

2. Data augmentation
   - The dataset we are using in this competition is the combination of training and validation set from VOC 2007
   - It contains only 5012 images in total. Furthermore, the labels are highly imbalanced
   - Doing data augmentation not only helps your model generalizing to testing data but also easing the training process

Hints

2. Data augmentation
   - Note that the bounding box coordinates have to be changed accordingly if the image was transformed

Hints

3. Training strategy
   - Check bugs
   - Be patient

Hints

4. Other object detection models
   - Feel free to try other object detection models
   - It is ok to read other's code on GitHub, but you have to implement it yourself
   - It's **not allowed** to load other's pre-trained model which was already trained on object detection task

Precautions

1. The final score will be only based on your ranking on private leaderboard (80%) and report (20%)
2. Training on the datasets not provided by us is forbidden
3. Loading the model pre-trained on ImageNet is allowed, while loading the model trained on object detection task is not allowed
4. Plagiarism gets you 0 point
5. Using ground truth to generate output will get you 0 point
6. Cloning codes from GitHub will you get 0 point

Competition Timeline

- Kaggle
- Timeline
  - 2025/10/16 (Thu) competition announced
  - 2025/11/11 (Tue) 23:59 competition due
  - 2025/11/16 (Sun) 23:59 report due
  - 2025/11/20 (Thu) top 3 team sharing