RSNA Pneumonia Detection Challenge

TEAM24

隊員：姚淨云、林孟學、楊亞澍

Outline

- Introduction
- Related work
- Proposed approach
- Experimental results
- Conclusions

GitHub: https://github.com/love124356/RSNA-Pneumonia-Detection-Challenge

Introduction

- Pneumonia Detection
  detect lung opacities on the Chest X-ray images
- Top 10 causes of death in the United States
  Top 3 causes of death in Taiwan

- Hard to Detect
- Use many modern object detection neural network architechtures
- Use many machine learning techniques

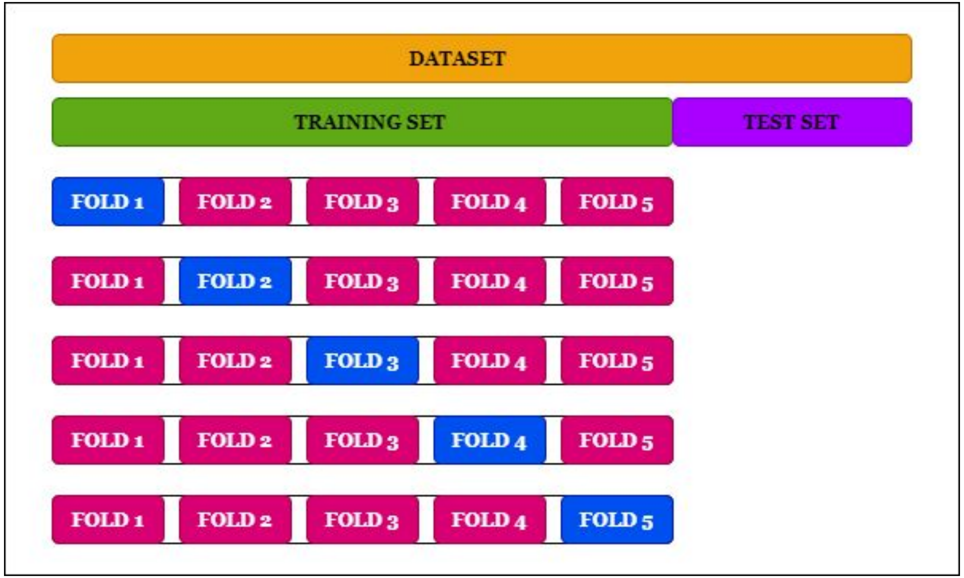
Related work

- Detection
- Two Stage Method - Faster R-CNN. Mask R-CNN
  - Choose a lot of proposals in the first stage, and detect the object from these proposals in the second stage.
  - **higher accuracy**
  - **time comsuming**
- One Stage Method - YOLOv5, RetinaNet
  - **faster inference speed**
  - **lower accuracy compare to two stage**
- Comparison
  - One stage methods perform well in many real time applications, but is not suitable in this task which needs **high accuracy**.

Related work

- K-Fold Cross Validation

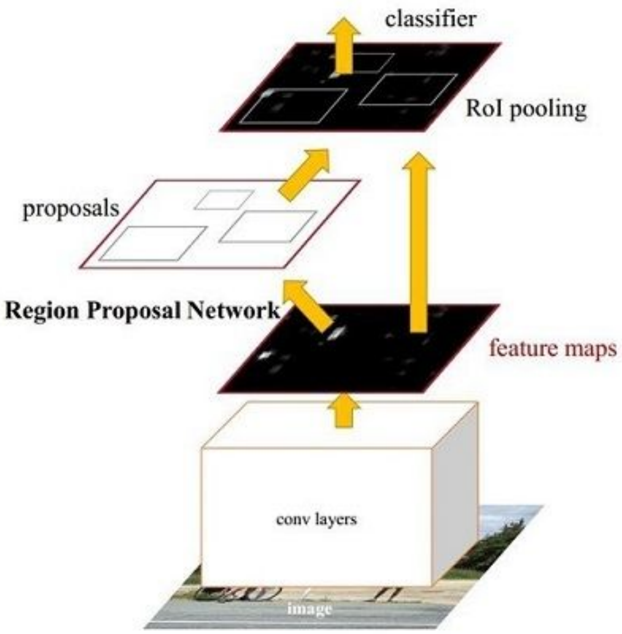
- Ensemble



https://medium.com/the-owl/k-fold-cross-validation-in-keras-3ec4a3a00538

Proposed approach

Faster R-CNN



The Faster R-CNN works as follows:

- The RPN generates region proposals.
- For all region proposals in the image, a fixed-length feature vector is extracted from each region using the ROI Pooling layer .
- The extracted feature vectors are then classified using the Fast R-CNN
- The class scores of the detected objects in addition to their bounding-boxes are returned.

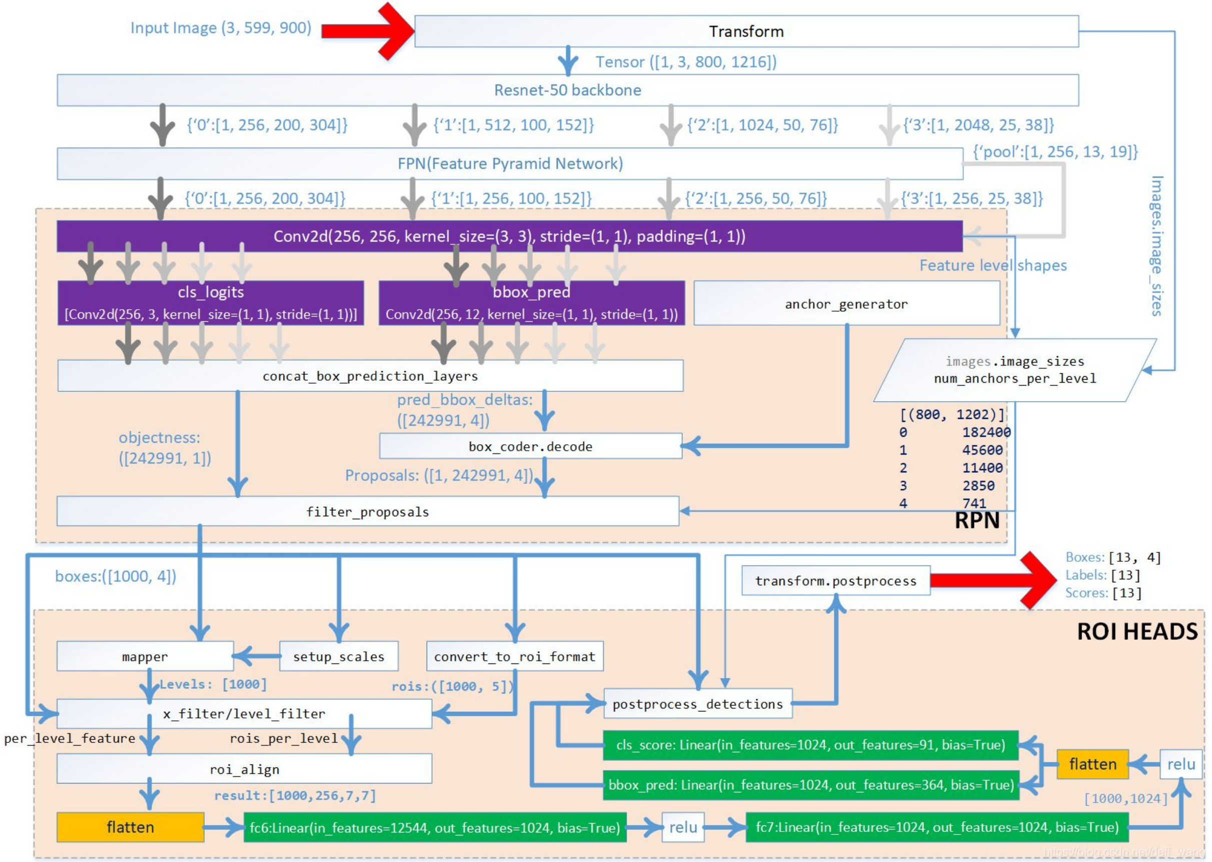
$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i,p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i,t_i^*).$$

Proposed approach

Faster R-CNN

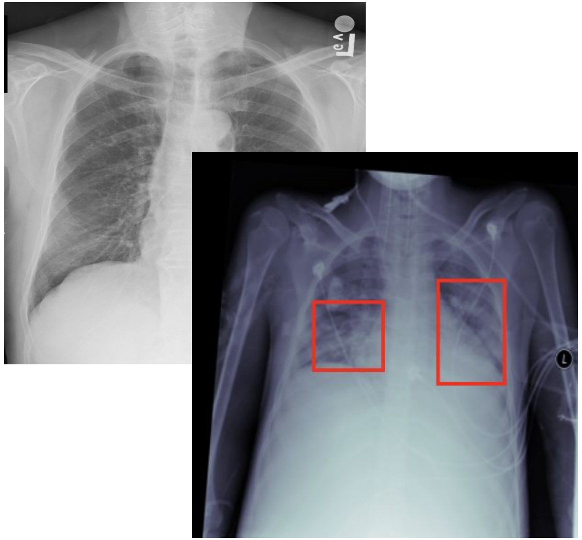
ResNet-50 FPN

1. Transform
2. Resnet-50
3. FPN
4. RPN
5. ROI



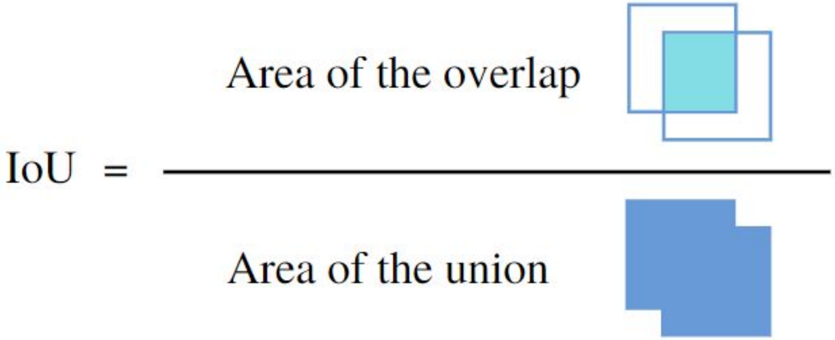
Dataset

- From the NIH CXR14 dataset using their original labels which were derived from radiology reports
  - 26684 training images (only 6012 images contain 9555 positive instances)
  - 3000 testing images
- Lables:
  - Not Normal / No Lung Opacity
  - Lung Opacity (only this label have bounging box)
  - Normal



Evaluation metrics

- Intersection over Union (IoU) - The measure is the degree of overlap between the two areas.
- By using IoU with a threshold to determine if the object is correctly detected



Evaluation metrics

- Average Precision (AP) - The average precision over thresholds.

  We also calculate false-negative if it indicates a ground truth box had no associated predicted box.

$$\text{precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives} + \text{False Negative}}$$

- Mean Average Precision (mAP) - The mean of APs over classes.

Experimental results

- Only tuned the hyper-parameters of YOLOv5 and Faster R-CNN.

Table 1. Comparison with related work.

	backbone	mAP
One-stage methods		
YOLOv5[3]	CSPDarkNet	13.948
RetinaNet[4]	ResNet-50	11.173
Two-stage methods		
Mask R-CNN[2]	ResNet-50	11.647
Faster R-CNN[1]	ResNet-50-FPN	**15.627**

Experimental results

- Compare mAPs by using different YOLOv5 architectures
- Change a training and validation data

Table 2. Compare mAPs by using different architectures

Model	Hyper-parameters	mAP
YOLOv5m6	300 epochs, 32 batch size, 608 x 608 image size, 0.3 conf. threshold	11.891
YOLOv5s6	300 epochs, 32 batch size, 608 x 608 image size, 0.3 conf. threshold	12.261
YOLOv5l6	300 epochs, 32 batch size, 608 x 608 image size, 0.3 conf. threshold	12.109
YOLOv5l6	300 epochs, 32 batch size, 608 x 608 image size, 0.3 conf. threshold, other training and validation data	**13.129**

Experimental results

- Compare to different batch size, image size of Faster R-CNN
- Change a training and validation data

Table 3. Models with different hyper-parameters

Model	Hyper-parameters	mAP
Faster R-CNN	50 epochs, 8 batch size, 300 x 300 image size, 1e-3 lr, 0.8 conf. threshold	13.114
Faster R-CNN	50 epochs, 8 batch size, 512 x 512 image size, 1e-3 lr, 0.8 conf. threshold	12.633
Faster R-CNN	50 epochs, 4 batch size, 300 x 300 image size, 1e-3 lr, 0.8 conf. threshold	13.807
Faster R-CNN	100 epochs, 4 batch size, 300 x 300 image size, 1e-3 lr, 0.8 conf. threshold, other training and validation data	**15.627**

Experimental results

- Faster R-CNN by using 5-fold cross validation and ensemble method

Table 4. Faster R-CNN using 5-fold cross validation and ensemble method

	mAP
Fold 0	15.627
Fold 1	14.590
Fold 2	14.587
Fold 3	14.756
Fold 4	8.831
+ *ensemble*	**16.336** (+0.709)

Conclusions

- Try many machine learning techniques to make result better:
  - data augmentation
  - k-fold cross-validation
  - ensemble method
- Get a result **16.336 mAP** with Faster R-CNN and ensemble method

Thanks for listening

Q & A