Covid-19 Detection with Chest Radiographs

Artificial Intelligence Group 12

Team Members

蔡博揚 R10922A15

- Faster-RCNN
- ViT
- ConvNext

徐榆婷 B07601004

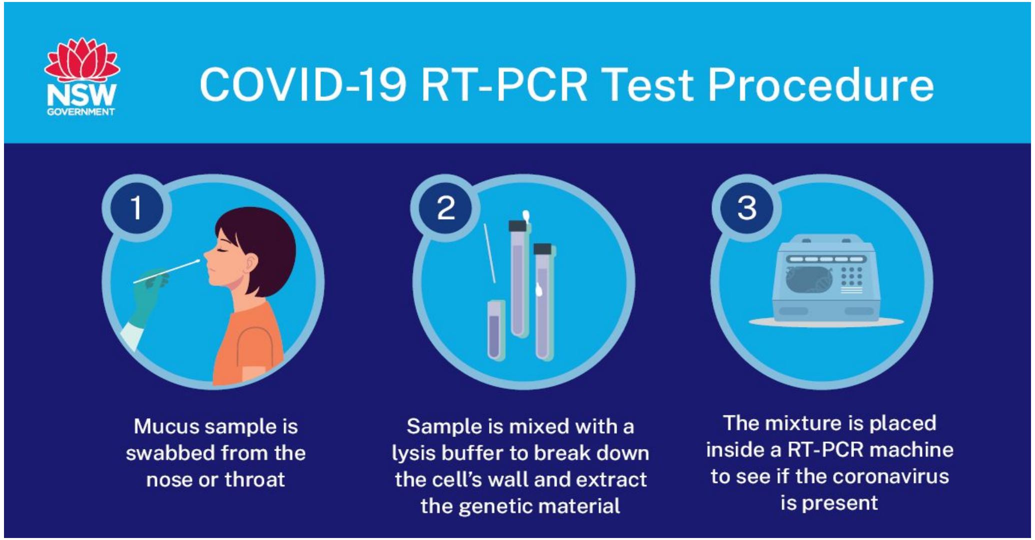
- YOLOv5
- EfficientNet

鄭婷卉 B07611046

- YOLOv5
- NASNet
- YOLOv5 - both tasks

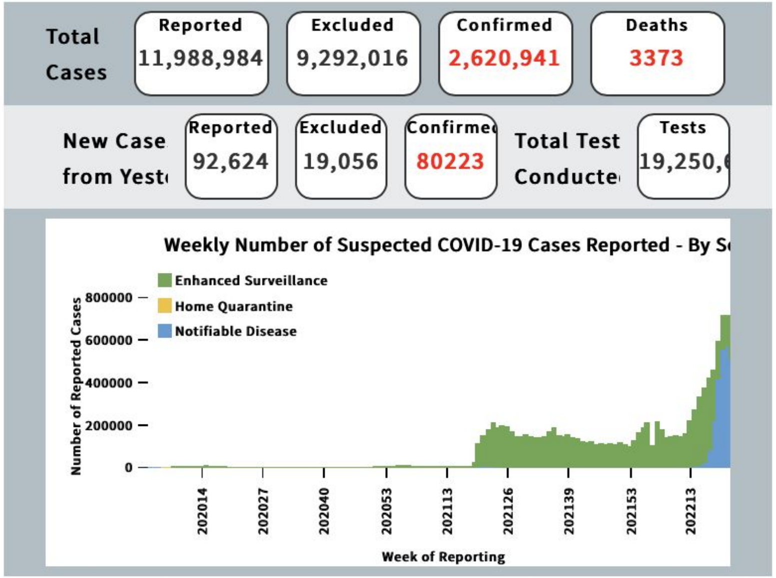
What is the problem?

- COVID-19 is an international health problem.
- It can be diagnosed by PCR but takes times.



Why is it important?

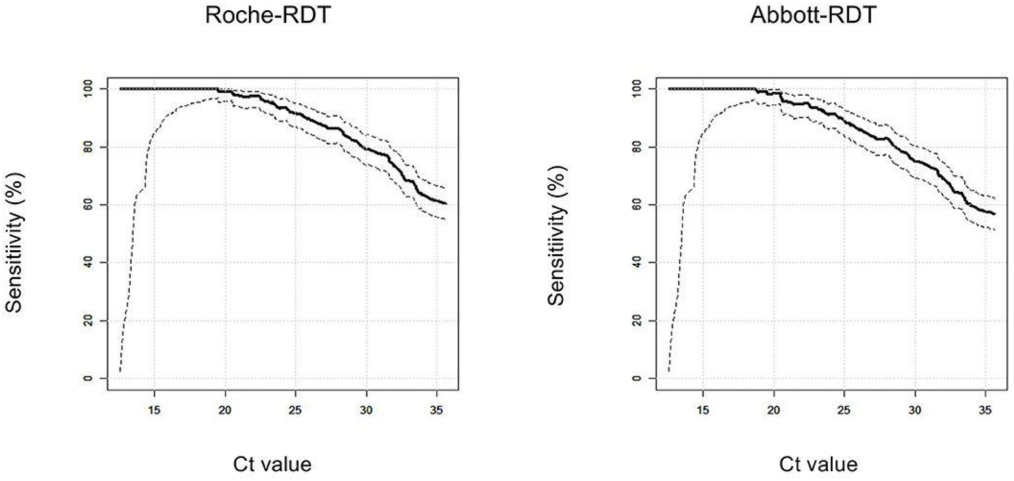
- Pandemic becomes more serious recently.
- Infectiousness peaks around one day before symptom and declines between 2-3 days before and 8 days after symptom. (CDC, US)



Updated time：2022-06-09 00:30

Why is it important?

- Rapid antigen tests accuracy
  - Roche/SD Biosensor: 65.3%
  - Abbot: 54.2%
- PCR accuracy: more than 95%



Solution overview - Chest radiographs detection

Therefore, we try to find another way to have a rapid covid test and can have higher accuracy.

And our solution is chest radiographs detection.

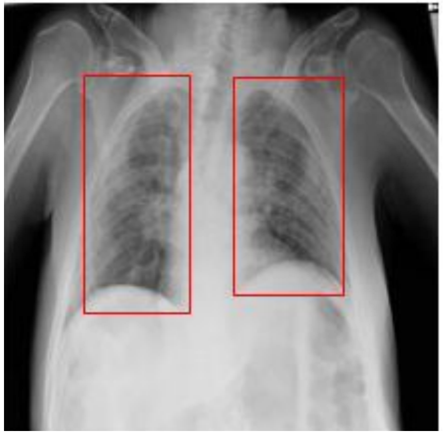
To do the chest radiographs, we need to do two tasks:

1. Localization
2. Classification

Solution overview - Chest radiographs detection

Localization

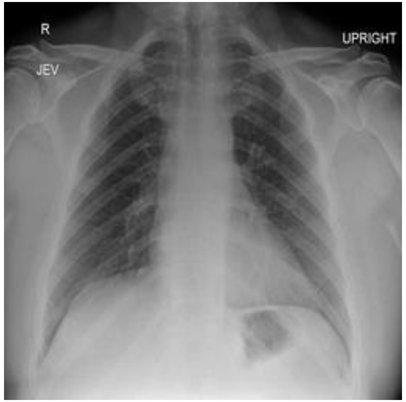
Chest X-ray image



Lesion detection

Solution overview - Chest radiographs detection

Classification



Negative

Typical

Indeterminate

Atypical

State of the art on object detection

- Deep CNNs can accurately and effectively differentiate normal and abnormal chest radiographs.
- R-CNN
- YOLO
- EfficientDet

Methodology

- Faster-RCNN
- Vision Transformer
- ConvNext
- Yolov5
- EfficientNet b0
- NASNet

Data resources | Datasets

- Society for Imaging Informatics in Medicine (SIIM)
- Image amount: 6335
- 2 Level
- 4 Categories in Study-Level

Image-Level	With BBbox	Total			
Cases	4295	6335			
Study-Level	Negative	Typical	Indeterminate	Atypical	Total
Cases	1676	2855	1049	474	6054

All the annotation have already been done by SIIM.

Tools / Frameworks used

- Pytorch
- Keras
- Huggingface
- Kaggle
- Albumentations

Lesion Localization

Faster-RCNN | Training Details

Pre-processing:

- Normalization
- Data Augmentation

Image size: 256 * 256

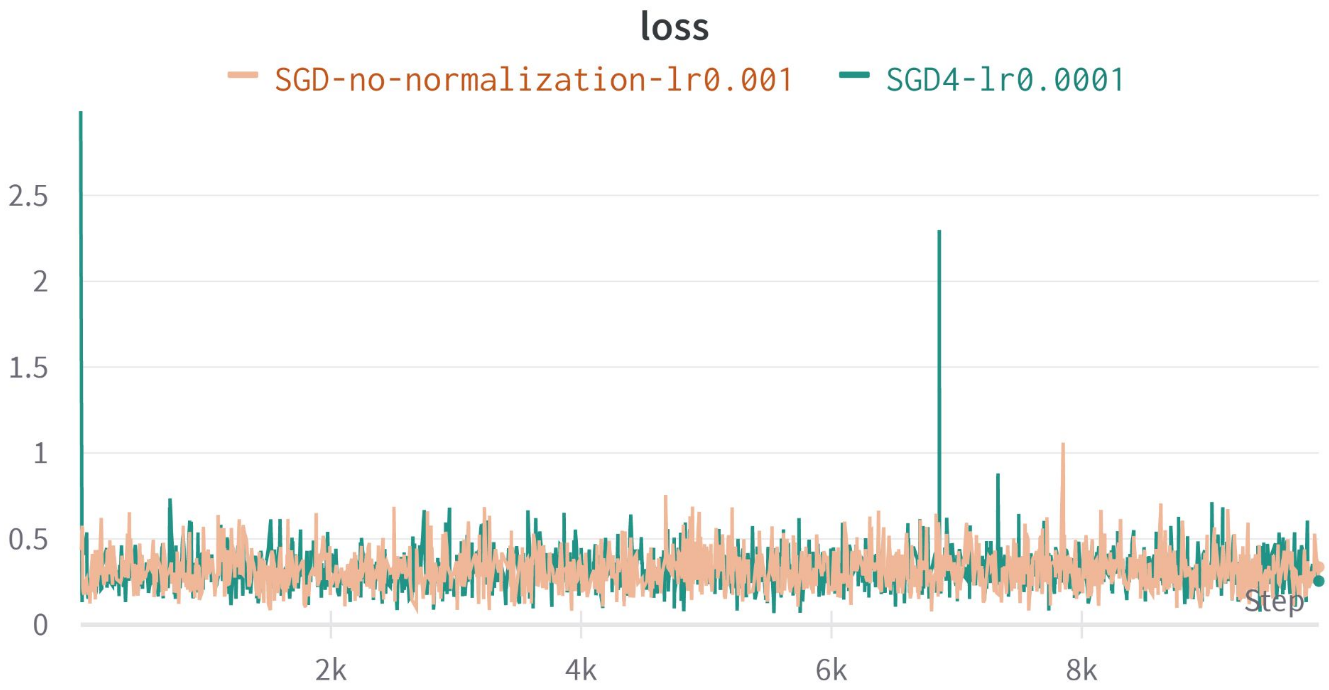
Batch size: 4

Epoch:5

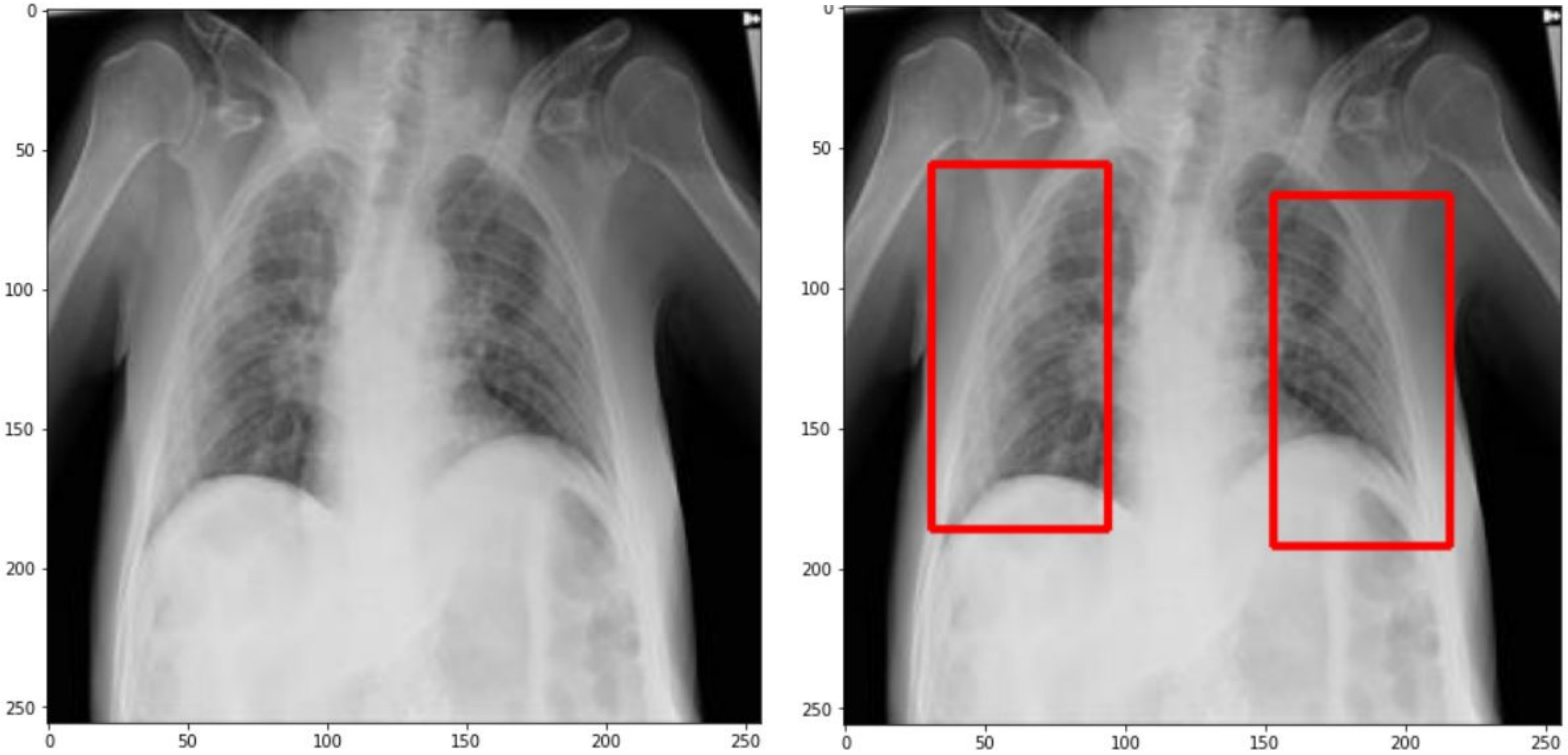
Optimizer: SGD

Initial learning rate: 0.001

Faster-RCNN | Model Performance



Faster-RCNN | Result



YOLOv5 | Training Details

Pre-processing:

- Normalization
- Data Cleaning

Image size: 256 * 256

Batch size: 16

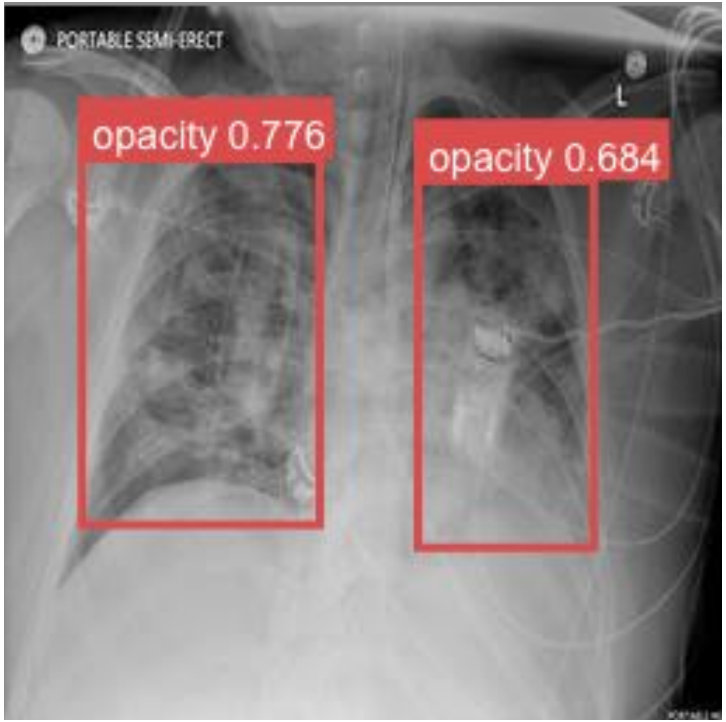
Epoch: 100

Optimizer: SGD

Initial learning rate: 0.01

YOLOv5 | Model Performance

YOLOv5 | Result



Observation

Comparing model performance of Faster-RCNN with YOLOv5, we can see that YOLOv5 get a higher performance for doing the localization.

Lesion Classification

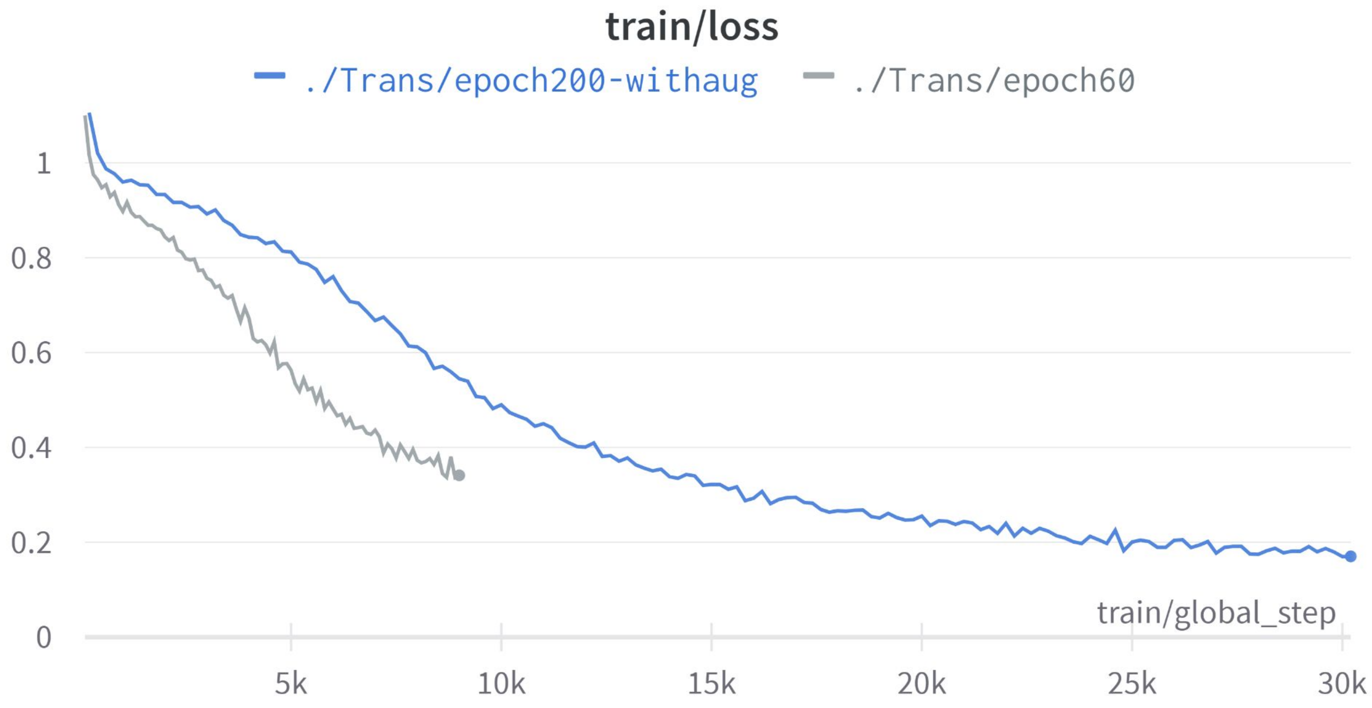
Vision Transformer | Training Details

Pre-processing:

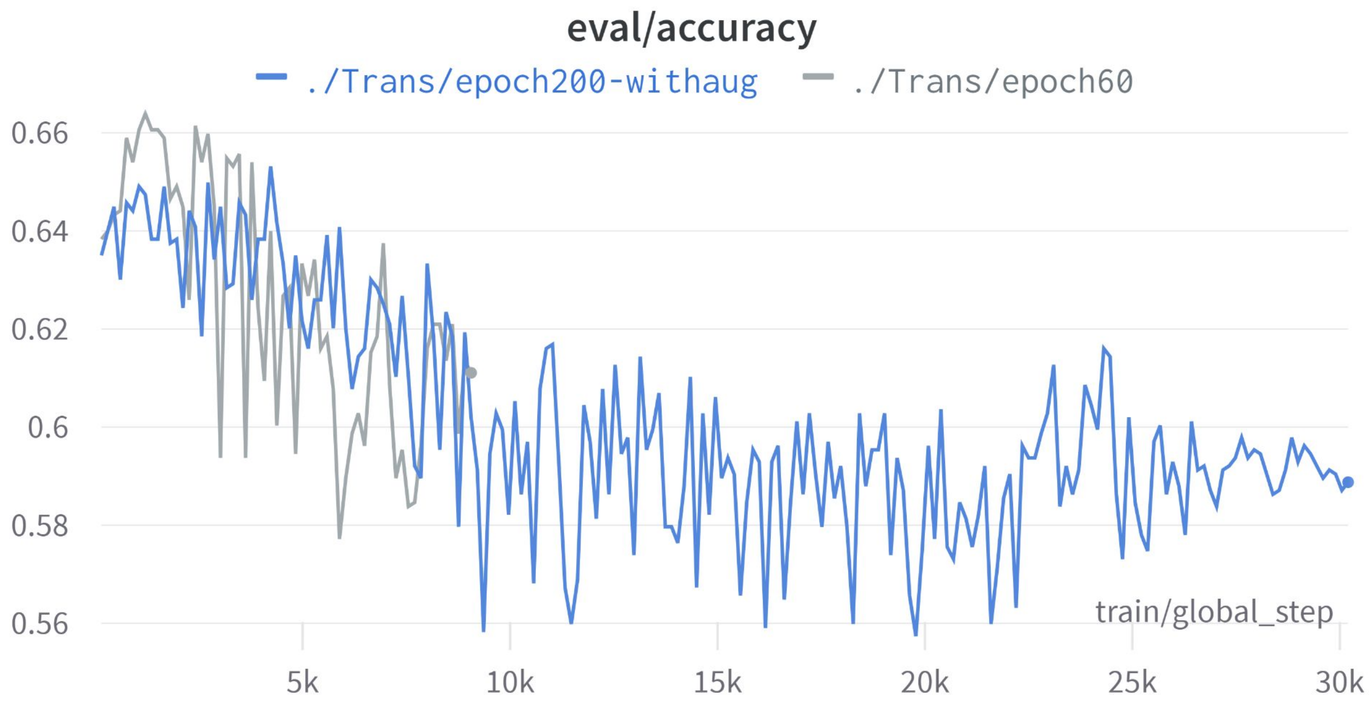
- Normalization
- Data Augmentation

Image size: 224 * 224
Batch size: 16
Epoch:200
Optimizer: Adam
Initial learning rate: 2e-5

Vision Transformer | Results



Vision Transformer | Results



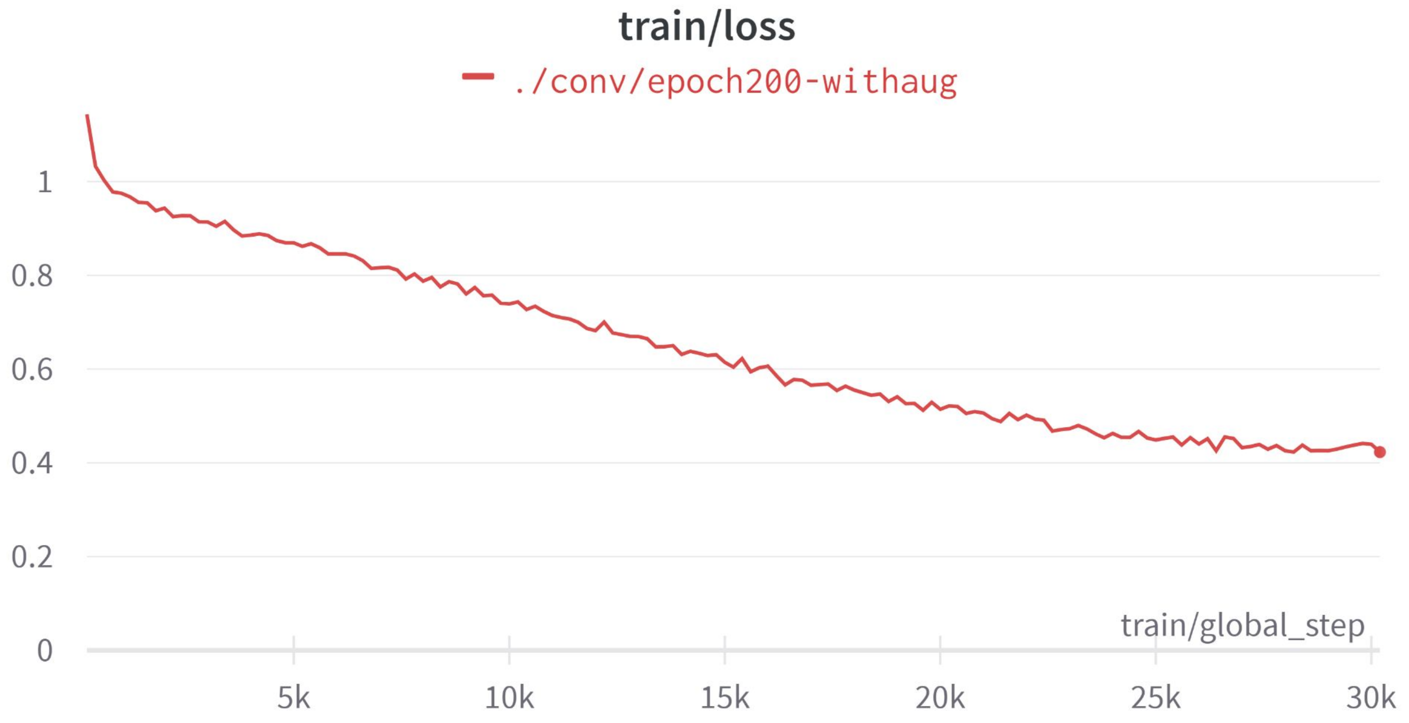
ConvNext | Training Details

Pre-processing:

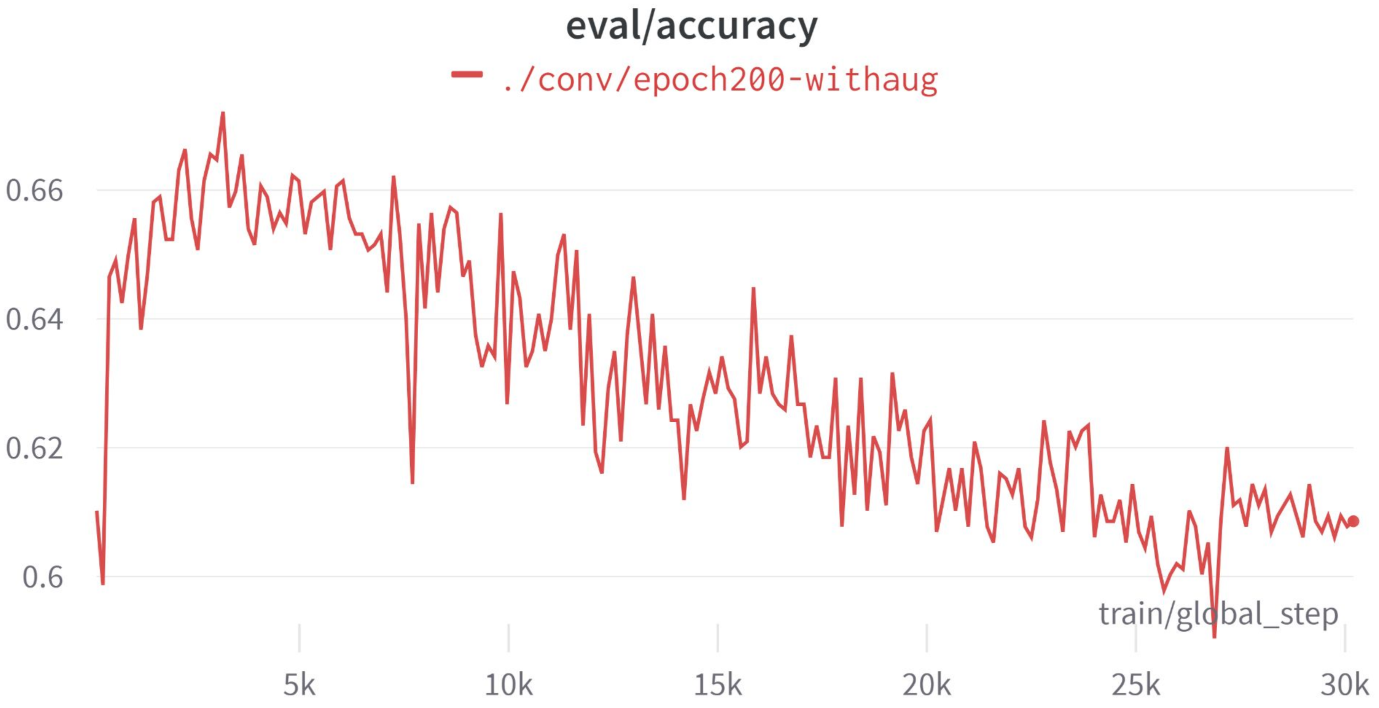
- Normalization
- Data Augmentation

Image size: 224 * 224
Batch size: 16
Epoch:200
Optimizer: Adam
Initial learning rate: 2e-5

ConvNext | Results



ConvNext | Results



EfficientNet b0 | Training Details

Pre-processing:

- Data Augmentation

Method

- K-fold

Image size: 224 * 224

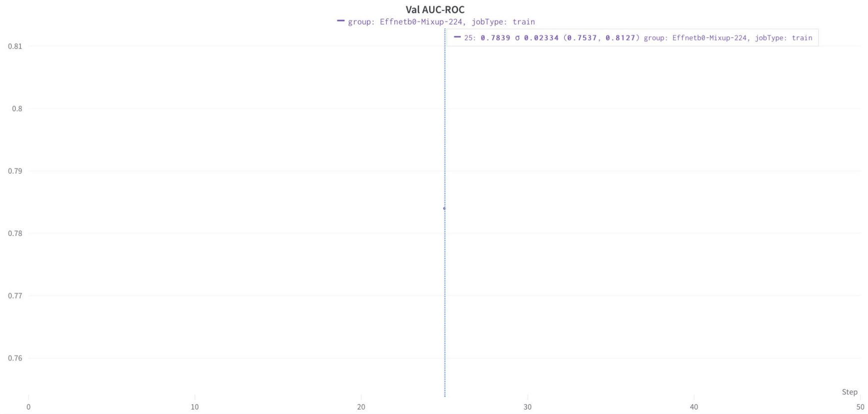
Batch size: 32

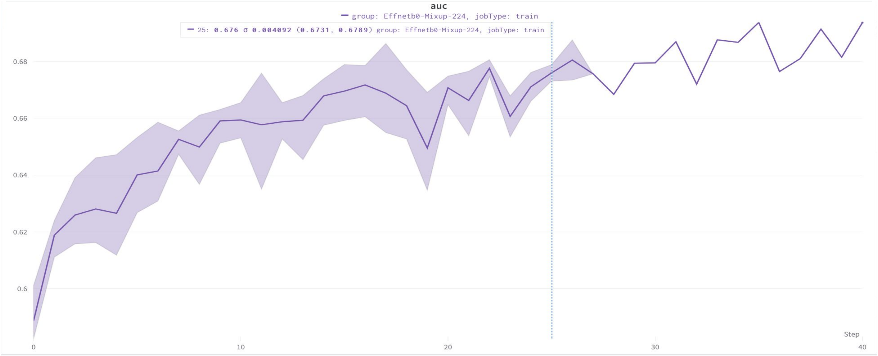
Epoch: 100

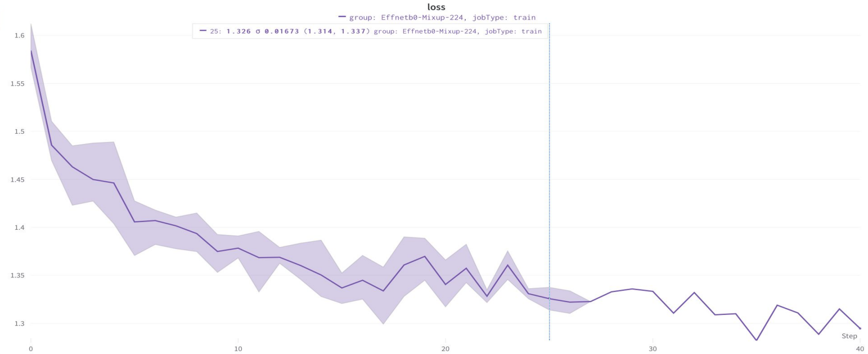
Optimizer: SGD

Initial learning rate: 0.01

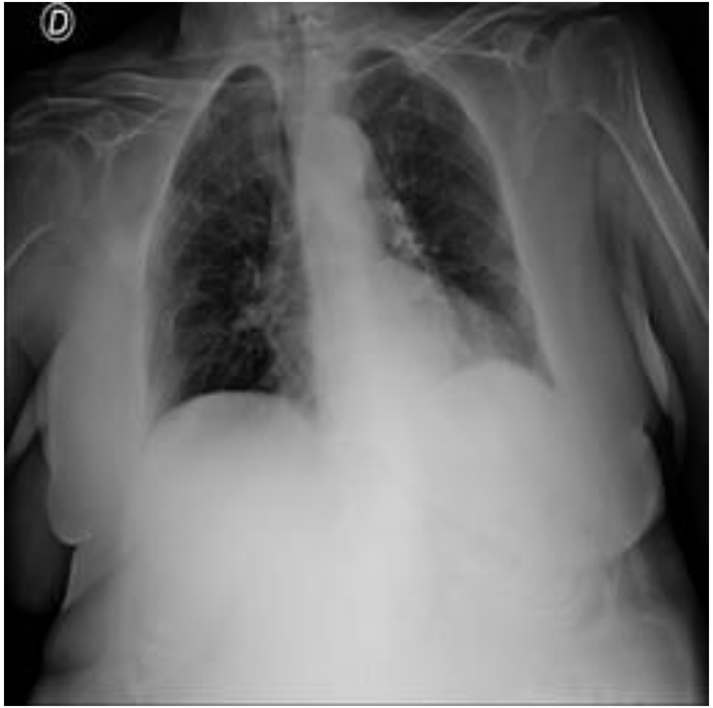
EfficientNet b0 | Model Performance

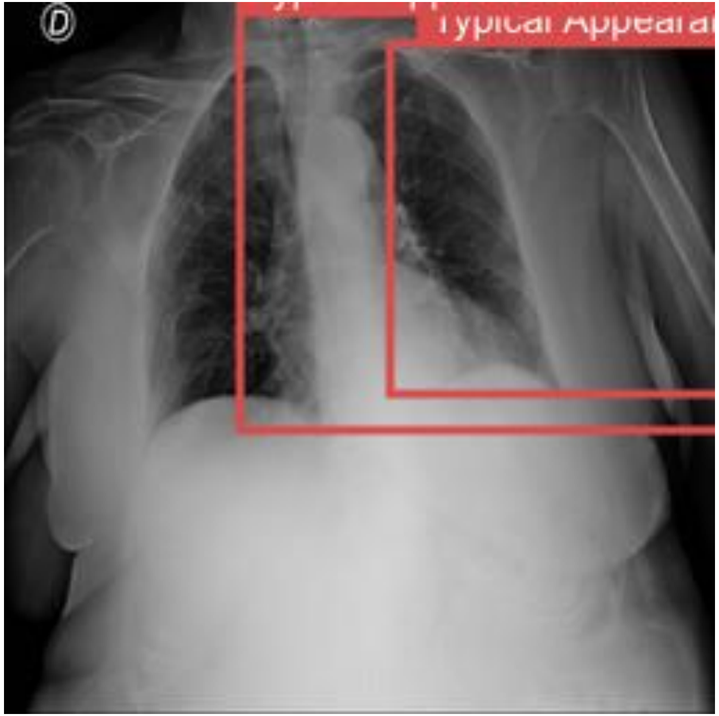






EfficientNet b0 | Results





NASNet | Training Details

Pre-processing:

- Data Augmentation

Image size: 224 * 224

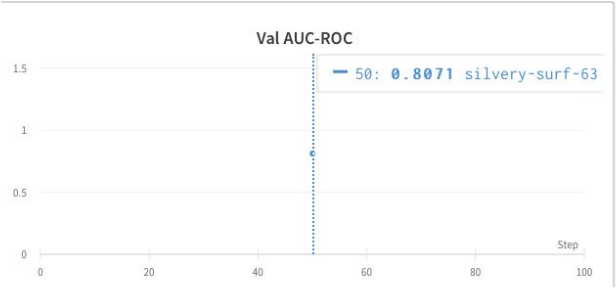
Batch size: 32

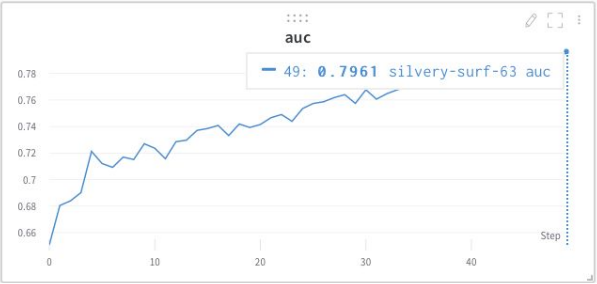
Epoch: 50

Optimizer: SGD

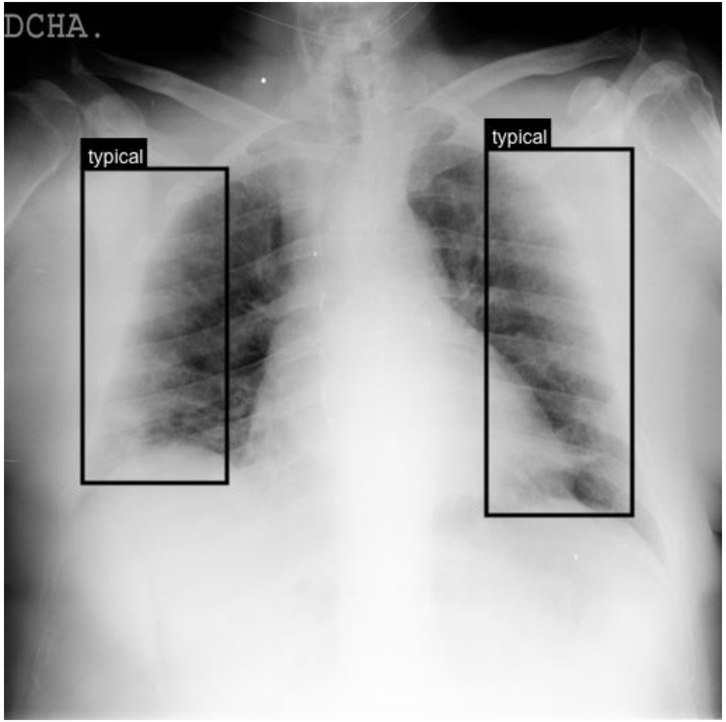
Initial learning rate: 0.001

NASNet | Model Performance





NASNet | Results



Observation

Comparing with model performance in different models, we can see that NASNet get a best performance for doing the classification. So, we choose NASNet as our final model to do this task in the end.

Do both tasks in one

Use YOLOv5 to do both tasks

Since we get not really good outcomes when we separate two tasks and use different models to deal with it, we are curious about whether combining localization and classification would be a better strategy.

As a result, we try to use one model, YOLOv5, to do both localization and classification.

YOLOv5 | Training Details

Pre-processing:

- Normalization
- Data Cleaning

Method

- K-fold

Image size: 256 * 256

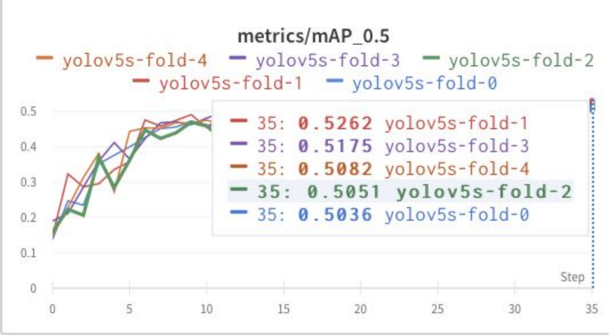
Batch size: 16

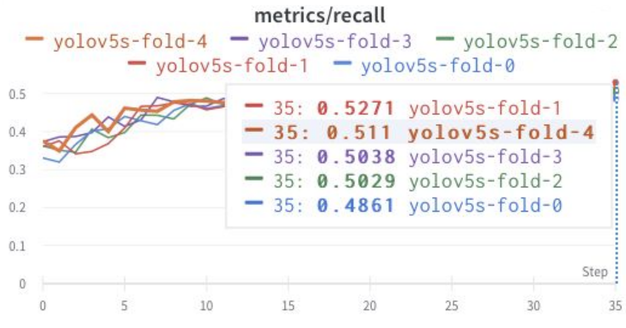
Epoch: 35

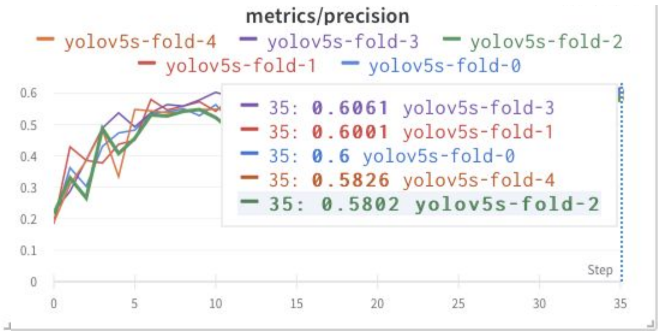
Optimizer: SGD

Initial learning rate: 0.01

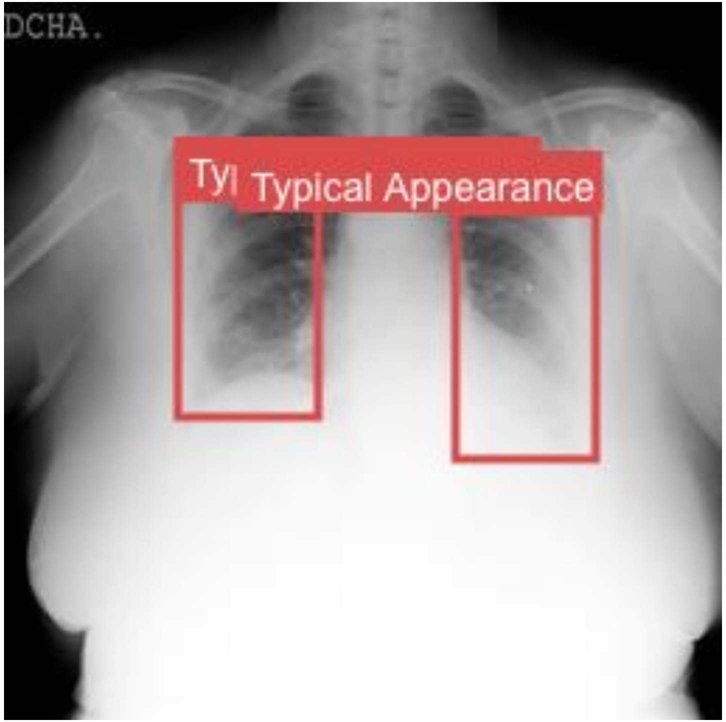
YOLOv5 | Model Performance





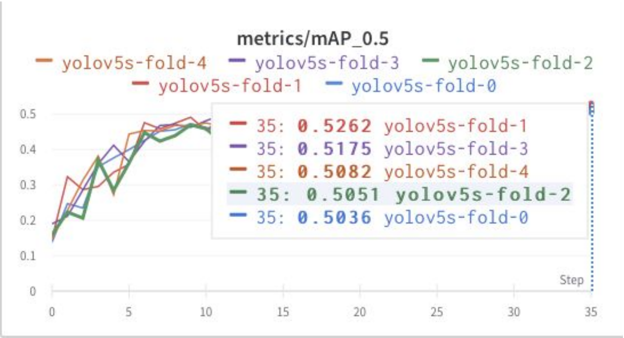


YOLOv5 | Results

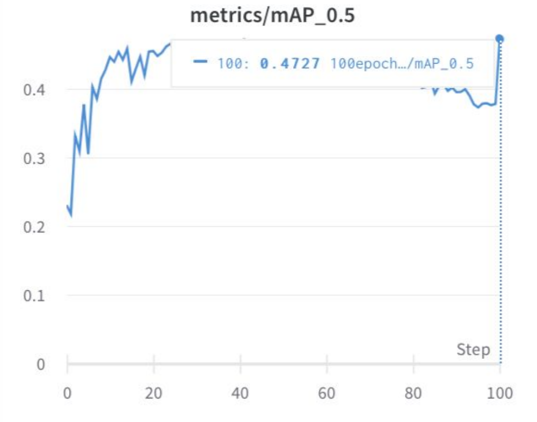


Observation

By the model performance and observing the testing results, we can conclude that it will get a higher performance when we use YOLOv5 to do both classification and localization.



YOLOv5 doing both task



YOLOv5 only for localization

Code links

https://drive.google.com/drive/folders/1Z5x55RbECAg4ecc-tbjJY6YZ5r9Ay37i?usp=sharing

Conclusion

We use two different ways to do the detection for the lesion. One is seperating this problem into localization and classification. we try two Faster-RCNN and YOLOv5 to deal with localization, and get a higher mAP_0.5 in YOLOv5, which is 0.4727. For classification. we try ConvNex, Vision Transformer, EfficientNet b0 and NASNet. NASNet get the highest AUC with 0.7961.

The other way that we solve this problem is combining localization and classification in one task. In the end, we get a highest performance in this way.

References

- https://www.kaggle.com/competitions/siim-covid19-detection
- https://www.kaggle.com/datasets/xhlulu/siim-covid19-resized-to-256px-jpg
- https://www.kaggle.com/datasets/xhlulu/siim-covid19-resized-to-224px-png
- https://www.kaggle.com/code/ayuraj/train-covid-19-detection-using-yolov5/notebook
- https://www.kaggle.com/code/ayuraj/train-study-level-classifier-w-b
- https://www.kaggle.com/code/piantic/train-siim-covid-19-detection-fasterrcnn
- https://huggingface.co/docs/transformers/main/en/model_doc/vit
- https://huggingface.co/docs/transformers/main/en/model_doc/convnext#transformers.ConvNextForImageClassification
- https://github.com/albumentations-team/albumentations

Answering questions

Q1: Isn't is much much harder to obtain a chest radiograph compared with rapid test or even PCR test? Could you elaborate more on the importance of your research?

A: Chest radiograph can be obtained in a few minutes while PCR could take hours or even days, and PCR test requires nurses to run the test station and to do the test, at a time when the nurse-patient ratios are already high, running a test station could only worsen the ratio. The precision of the rapid test is not as good as the chest radiograph detection.

Our purpose is to reduce the waiting time of the diagnosis as a result of this overwhelming medical system, so a covid-19 diagnosis with a chest radiograph might be a solution to it.

Answering questions

Q2: How do you get the x-ray image dataset? How many images for training / valid / tests?

A: The image datasets are from SIIM(Society for Imaging Informatics in Medicine), and there are 6054 images for training, and 1214 images for testing.

Answering questions

Q3: Why does your group choose these techniques? Do you have the comparison with the baseline?

A: In order to find the location of lesion and classify different syndromes, we choose CNN models. CNN models are state-of-art in the domain of computer vision.

Our dataset comes from a competition on kaggle. We can compare our result to other competitors.

Answering questions

Q4: How do you do the data augmentation?

A: In Pytorch torchvision, it provides transformation for images which are tensors or numpy arrays. For bounding box, we use Albumentation library.

Answering questions

Q5: 光是 dataset 標註就需要 domain knowledge，如何要跟 PCR 陽性的輕症(或無症狀)患者做連結?

A: The label was done by experts, and they are being labeled into 4 categories which are, "negative", "typical", "indeterminate", and "atypical".

Answering questions

Q6: 從loss來看，從頭一直到最後都在特定範圍持續跳動，應該沒有收斂啊?

A: This loss is for each mini-batch, so it looks oscillating. It actually converages in terms of epochs. (If I showed you training loss for each epoch, it would be a horizontal line.)

Answering questions

Q7: For optimizing RCNN, is SGD the best optimizer? Is there any reason that it's better than Adam or other more complex optimizers?

A: For RCNN, we tried SGD and Adam on RCNN, the mAP for Adam are almost 0, and SGD is slightly better (0.02), so we chose SGD.

For yolov5, we tried SGD and Adam on yolov5, the results are similar but SGD is slightly better on the mAP scores, so we chose SGD.

Answering questions

Q8: 請問你的結果會比醫師診斷更準確嗎?

A: We are not doctors, so we don't have the chance to compare our results with a doctor's diagnosis. However, it would be great to compare, so we'd know where to improve. Also, even if our results are not as good as doctors' diagnoses, it could still help by localizing the lesion area, so the doctors can make diagnoses more quickly.

Answering questions

Q9: Can it still detect if we only have mild symptoms?

A: Yes, our model will classify the located lesions into 4 categories which are, "negative", "typical", "indeterminate", and "atypical".

Answering questions

Q10: 抽菸跟沒抽菸的肺有分開訓練嗎?

A: We don't know if the dataset contains smoking data or not.

Answering questions

Q11: 有肺部舊疾會影響到判斷嗎？

A: We don't know for sure, but the dataset was collected with all kinds of chest radiographs, so it shouldn't be a problem.

Answering questions

Q12: How long does it take to train a model, and how about the accuracy?

A: For yolov5, we trained around 6 hours and its mAP_0.5 is 0.4727

For Faster RCNN, we trained around 1 hours and its mAP_0.5 is 0.02

THANK YOU

- Team Members
  - Name/ID/Photos with individual responsibilities/contributions
  - Collaboration/communication mechanism
- What is the problem?
- Motivation: why is it important?
- What's the state of the art? i.e. survey of related work
- Solution overview: what's your big idea?
- Technical details: How did you solve the problem?
  - methodology/algorithms (w/ links to your codes)
  - data resources w/ annotation, pre-processing, aggregation
  - tools or frameworks used
- What are the results (experimental outcome, observation/analysis)?
- References
- Answers to questions during the final presentation

Data resources | Pre-processing

- Faster-RCNN
  - normalization
  - data augmentation
- YOLOv5
  - the content of annotations text file needs to do normalization
  - mosaic augmentation is already invovled inside the model
- NASNet
  -

Methodology | YOLOv5

- An object detection system targeted for real-time processing

Pros:

  - Fast Speed
  - Generalize image better

Cons:

  - limited detection

Methodology | NASNet

Faster-RCNN | Methods

- An object detection system using Regional Proposal Network

Pros:

- More Accurate

Cons:

- Slow Speed

Methodology | K-fold