Distinguishing Pneumonia with X-ray of Chest via Deep Learning

made by Xiao-Ying Zhuang

Outline

Summary

- Problem
- Solution
- Target
- Flow Diagram

Training

- Procedure
- Data Source
- Preprocessing
- Training Models
- Criteria
- Comparison
- Result

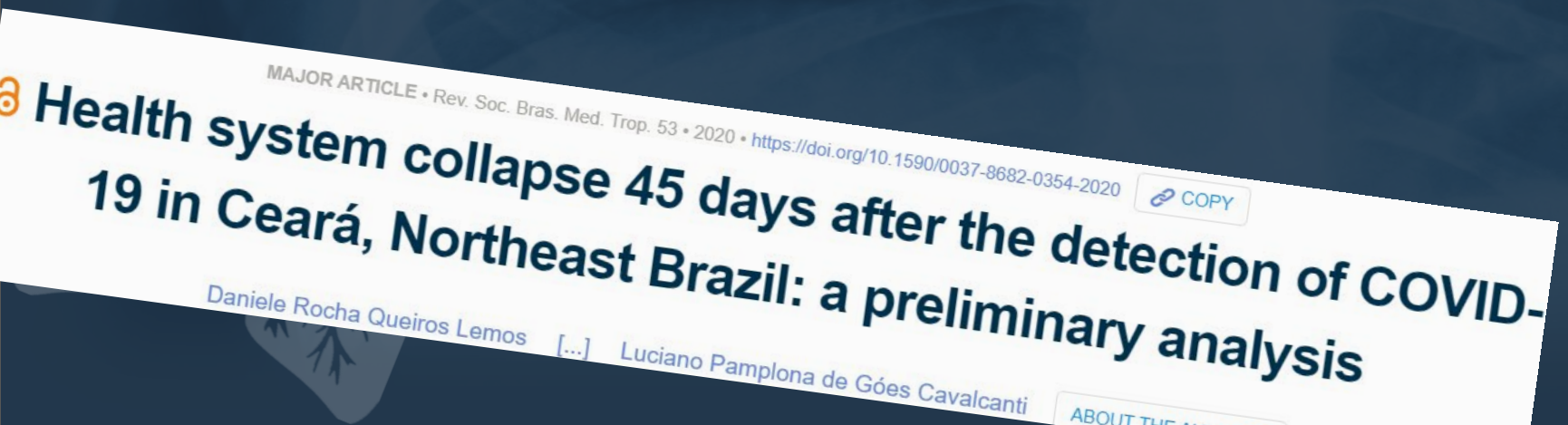
Application

- Demo (flask)
- Demo (tkinter)
- Conclusion
- Future Outlook
- Reference

Problem

The outbreak of the Covid-19 in late 2019 triggered a global epidemic of infectious diseases and paralyzed the global healthcare system.

When a patient with suspected Covid-19 is sent to the hospital, it is necessary to quickly judge its severity so that the amount of medical treatment can be properly allocated to prevent severe patients from running out of resources.



Solution

In practice, the diagnosis of pneumonia depends on the following things:

- ▲ Clinical symptoms (cough, sputum, fever, pain)
- ▲ Medical Images (Abnormal shadowing in the chest X-ray Image)

However, the interpretation of chest X-ray images will be more time-consuming. Besides, the more severe the lesion is, the easier it is to interpret. In other words, if the lesion is very small, there may be interpretation errors. And poor exposure imaging may also affect interpretation.

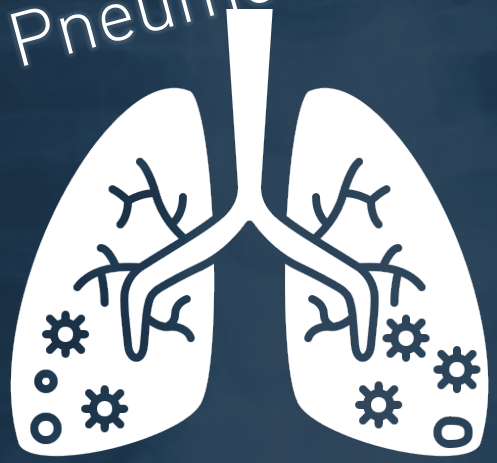
By using deep learning model (CNN), the problems mentioned above can be solved quickly and precisely. Because CNN model can“read” X-ray images pixel by pixel, which is almost impossible to human beings.

X-ray Image → Prediction Model → Normal/ Pneumonia

Target

X-ray Image

Pneumonia...

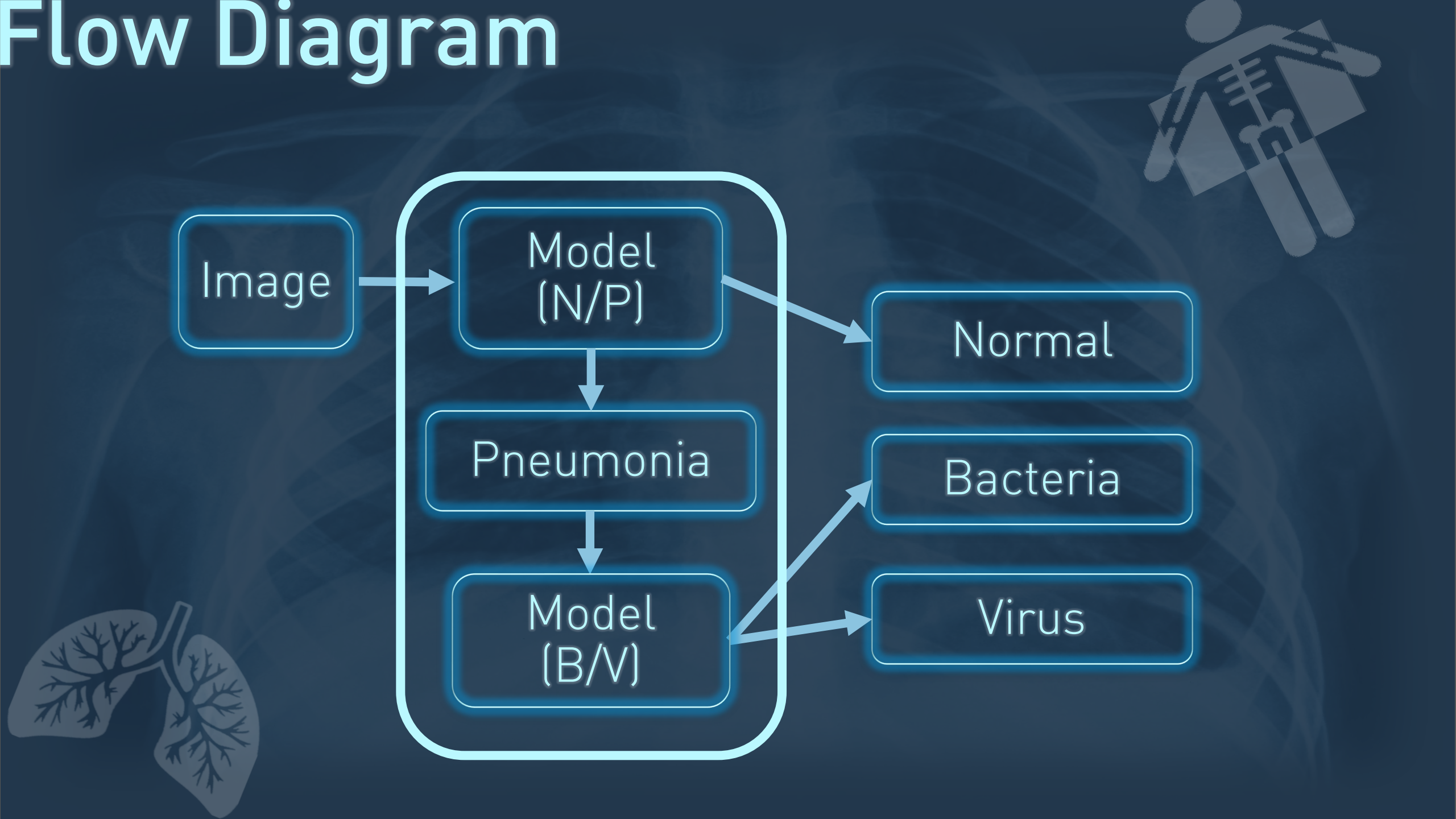


Becteria

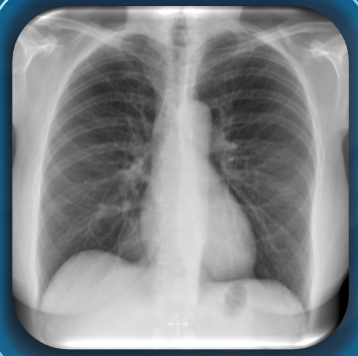
Virus

Normal!

Flow Diagram



Precedure



Preprocessing

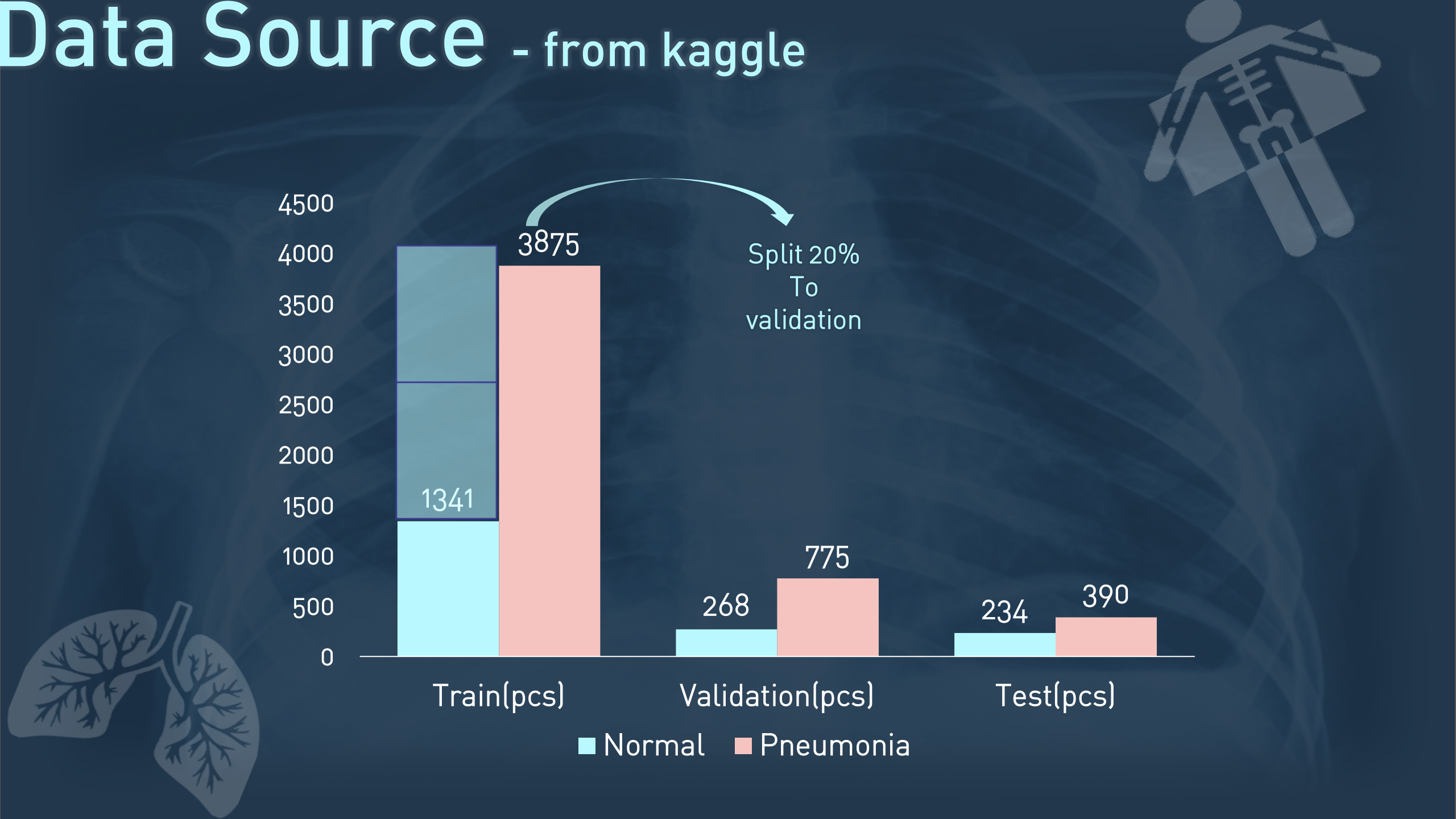
Training

Label
- Normal
- Pneumonia
  - Virus
  - Bacteria

- Rotation_range
- Width_shift_range
- Height_shift_range
- Zoom range
- resize

- weight
- Epoch
- batch size
- EarlyStopping
- ModelCheckpoint

Data Source - from kaggle



Preprocessing

- rotation_rang = 30
- width_shift_range = 0.05~0.2
- height_shift_range = 0.05~0.2
- zoom_range = 0.2
- horizontal_flip = false

Training Models

- Weight = {0:3,1:1}
- Input_shape = 128~666
- Epoch = 30~45
- Batch size = 32~64
- Model_monitor = Recall
- ReduceLROnPlateau
  - Factor = 0.2~0.3
- Early_stopping =
  - monitor = loss
  - Min_delta = 1e-5~1e-2

Criteria

		Actual: True	Actual: False
Prediction	True	TP	FP
Prediction	False	FN	TN

- Accuracy = (TP + TN)/(TP + FP + FN + TN)
- Precision = TP / (TP + FP)
- Recall= TP / (TP + FN)
- F1 score = 2 / (Precision^{-1} + Recall^{-1})

Training Models - Self-build CNN

- Conv2D (32 ,(3,3))
- Activation : relu
- MaxPool2D (2,2)
- Conv2D (64 ,(6,6))
- Activation : relu
- MaxPool2D (2,2)
- Conv2D (128 ,(9,9))
- Activation : relu
- MaxPool2D (2,2)
- Flatten
- Dense(64)
- Activation : relu
- Dropout 50%
- Dense(1)
- Activation : sigmoid

```
model = Sequential()
model.add(Conv2D(32, (3,3), input_shape = img.shape))
model.add(Activation("relu"))
model.add(MaxPool2D(pool_size=(2,2)))

model.add(Conv2D(64, (6,6)))
model.add(Activation("relu"))
model.add(MaxPool2D(pool_size=(2,2)))

model.add(Conv2D(128, (9,9)))
model.add(Activation("relu"))
model.add(MaxPool2D(pool_size=(2,2)))

model.add(Flatten())
model.add(Dense(64))
model.add(Activation("relu"))
model.add(Dropout(0.5))
model.add(Dense(1))
model.add(Activation("sigmoid"))
```

Training Models - Keras applications

- Both ResNet50V2 and ResNet152V2 are versions of ResNet that have improved the vanishing gradient problem when dealing with deep networks.
- InceptionV3 combines multiple convolution operations (such as 1x1, 3x3, and 5x5 convolutions), having pooling layers of different sizes and types to improve model efficiency.
- MobileNet is a lightweight network architecture using depthwise separable convolutions that can run on mobile devices with limited hardware.
- EfficientNetV2S is a version of EfficientNet, which is mainly designed based on the combination of network depth, width and resolution.

Ensembling

X-ray data

Self-build CNN

InceptionV3

ResNet152V2

EfficientNetV2S

X-ray

N / P Prediction

Normal

Congratulations!!

Pneumonia

Self-build CNN

Mobile Net

ResNet50V2

ResNet152V2

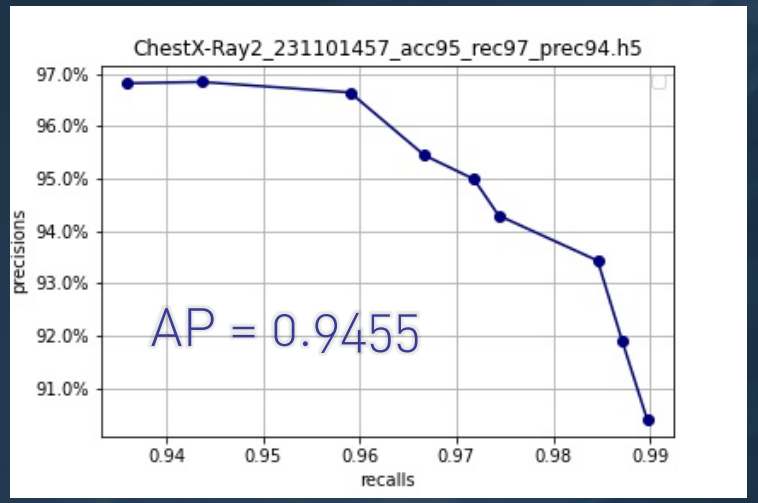
Pneumonia X-ray

B / V Prediction

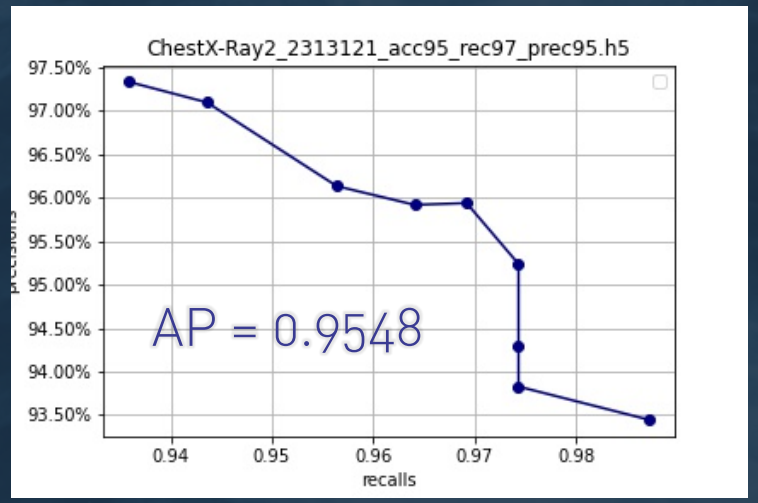
Bacterial pneumonia

Viral pneumonia

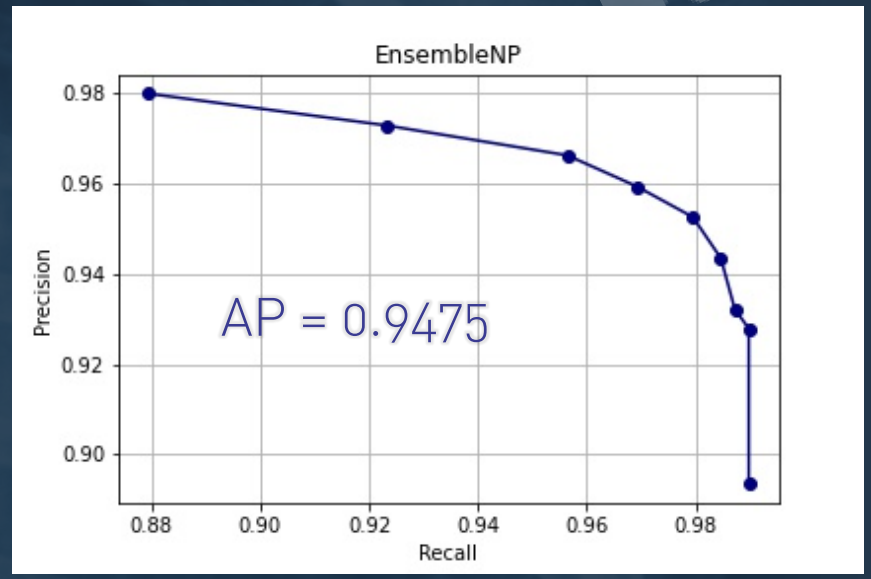
Comparison – PR curve



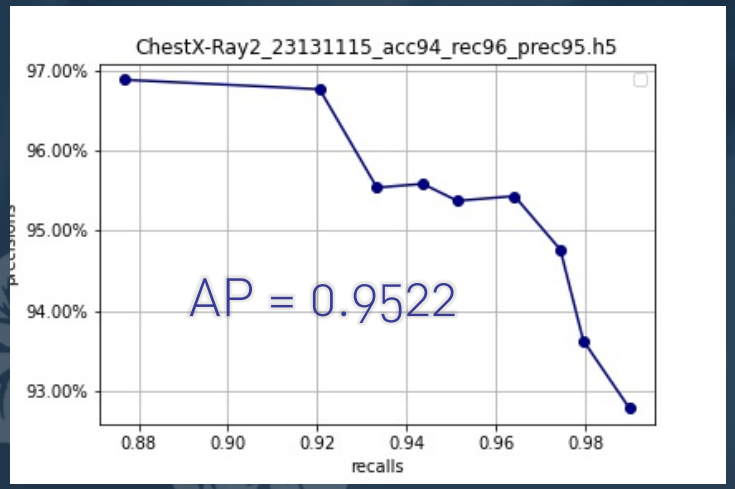
inceptionV3



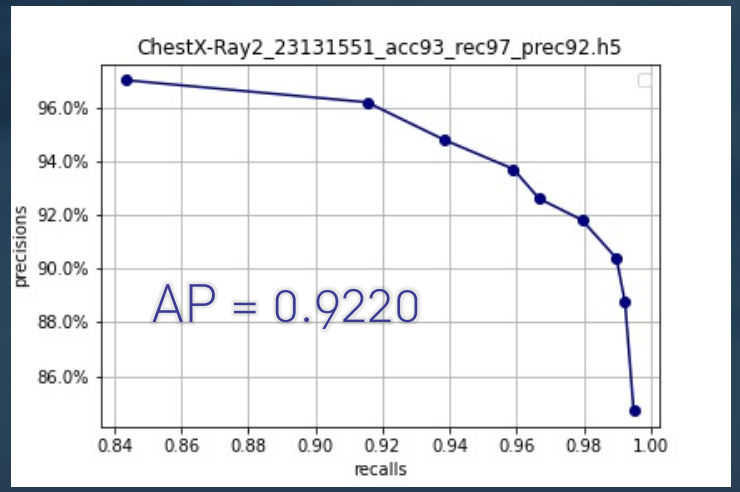
EfficientNetV2S



Ensembled

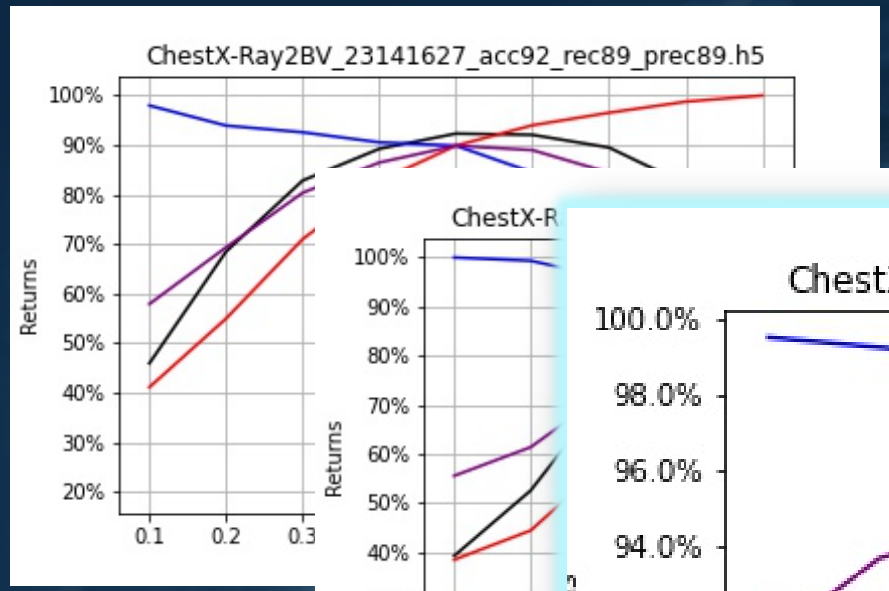


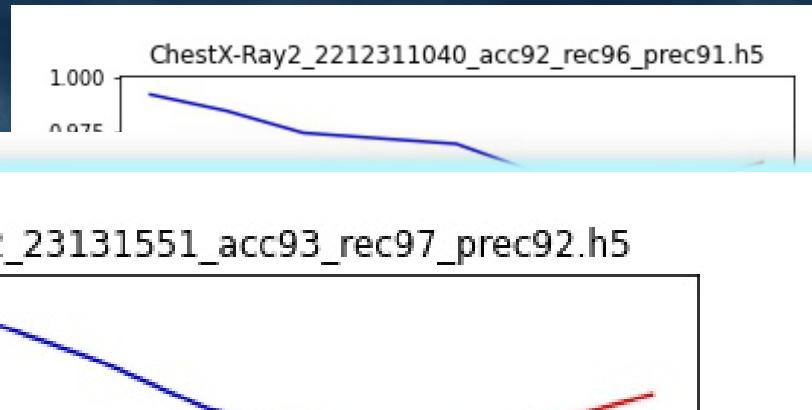
ResNet152V2

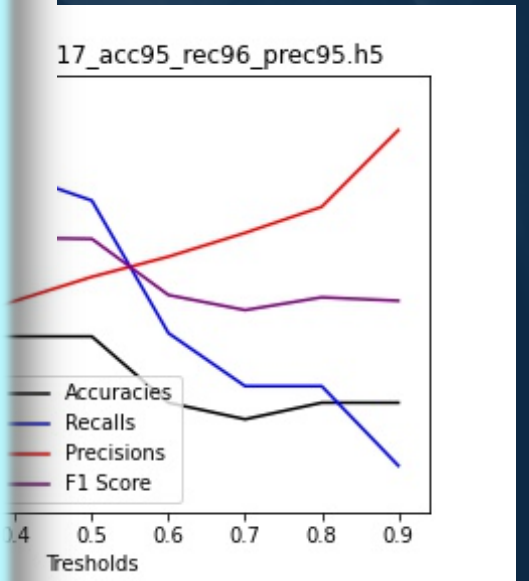


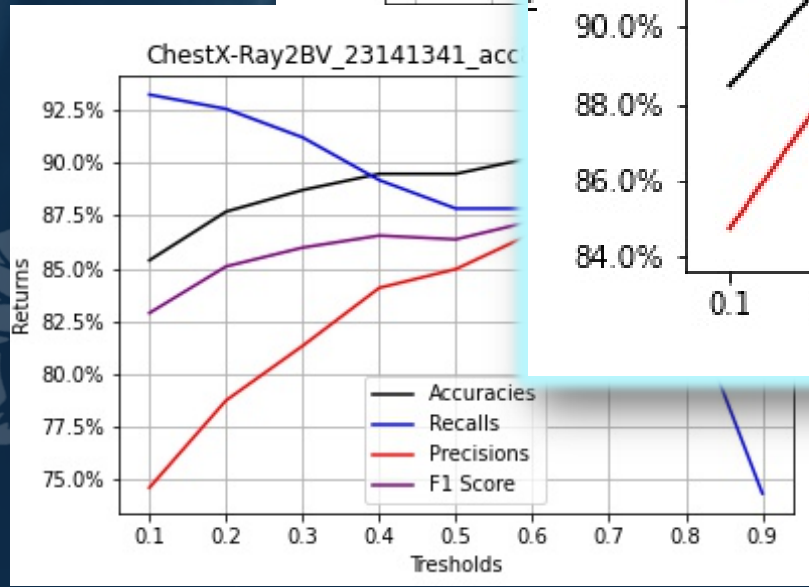
Self-build CNN

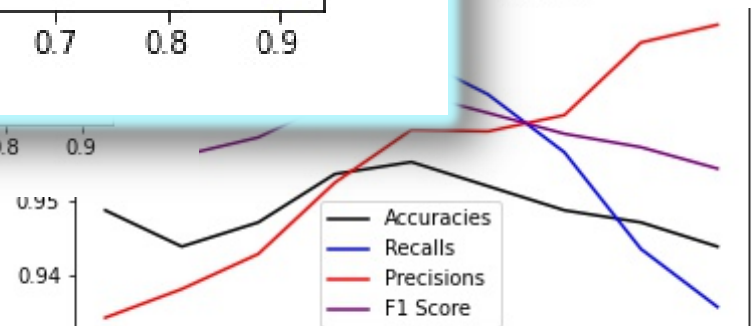
Comparison – best threshold finding





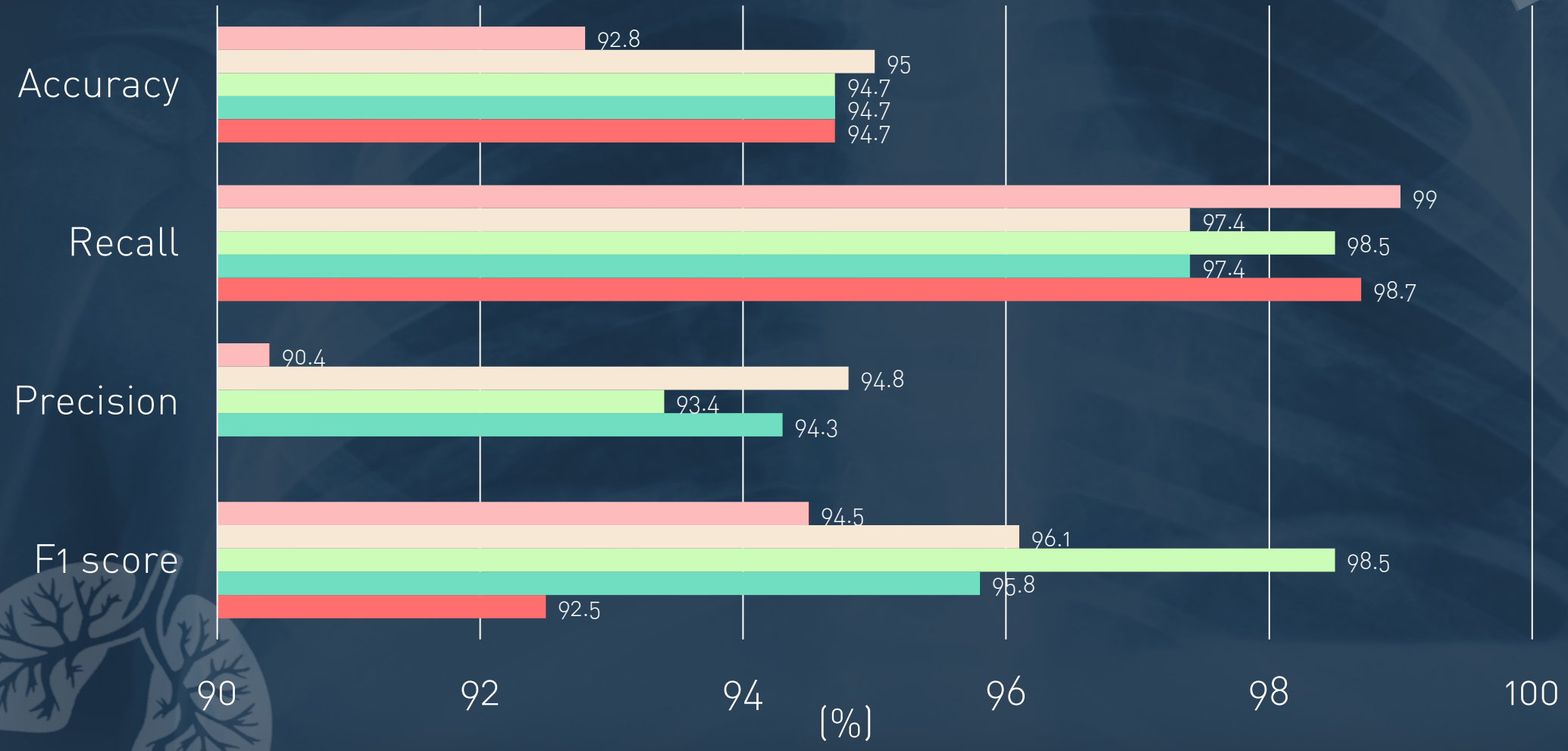






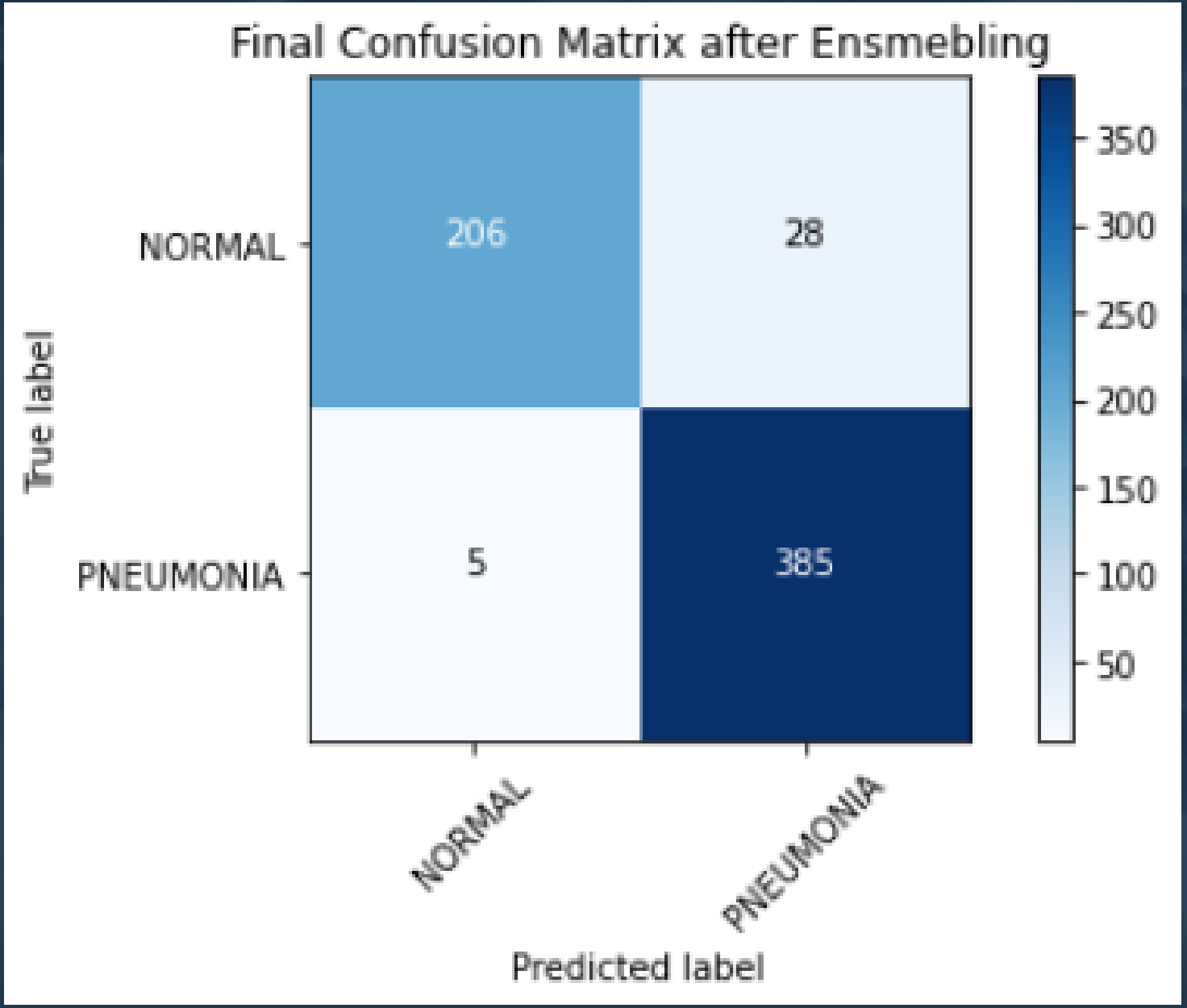
Result

Pneumonia / Normal



Result

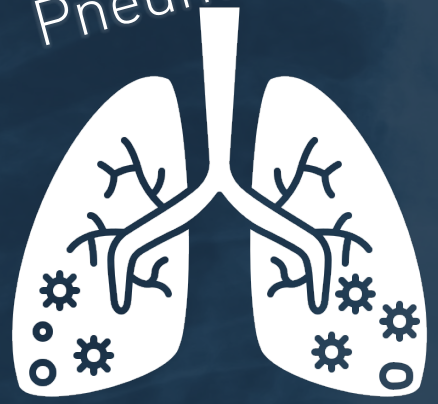
Pneumonia / Normal



Result

Pneumonia / Normal

X-ray Image



Pneumonia...

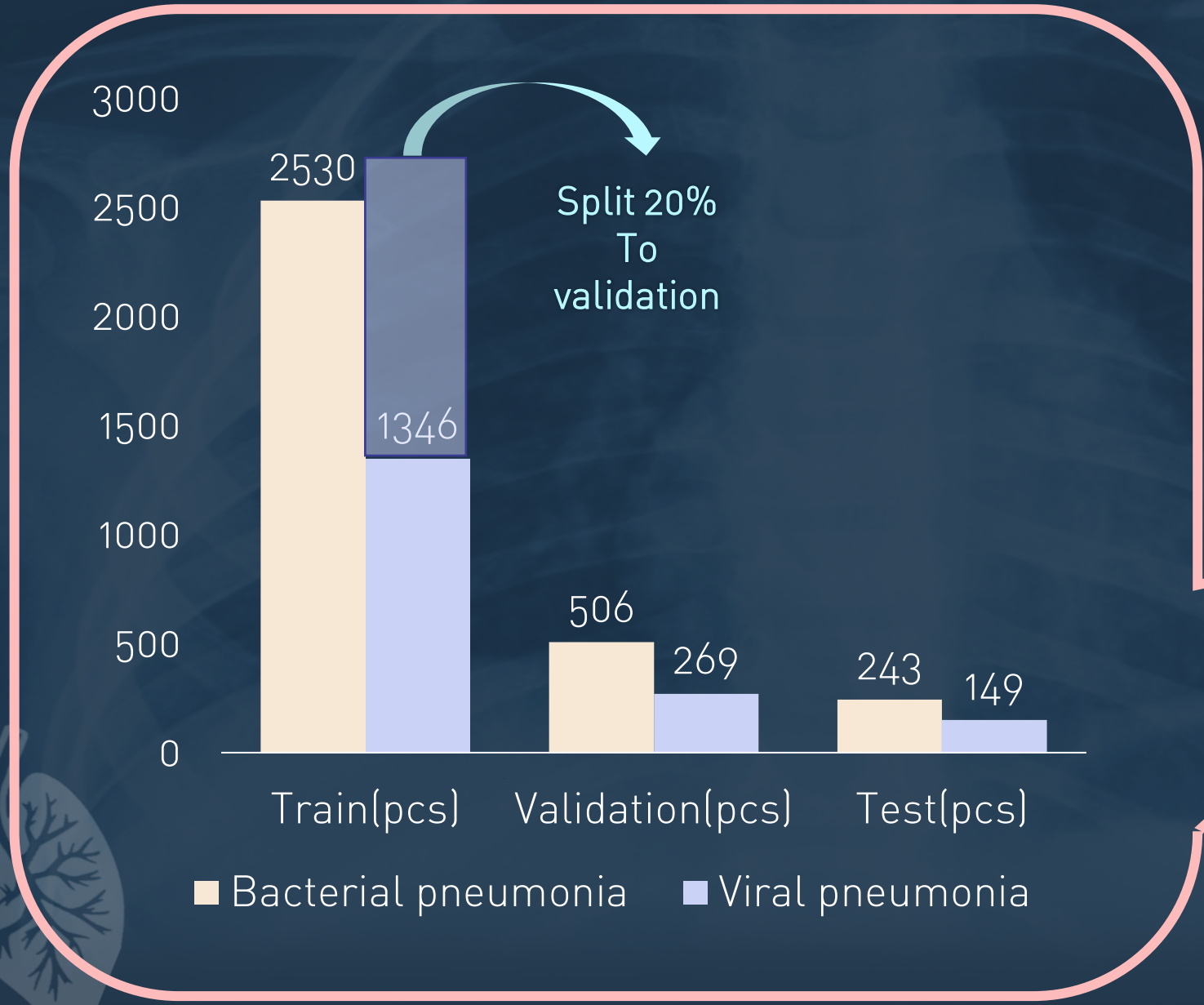
?

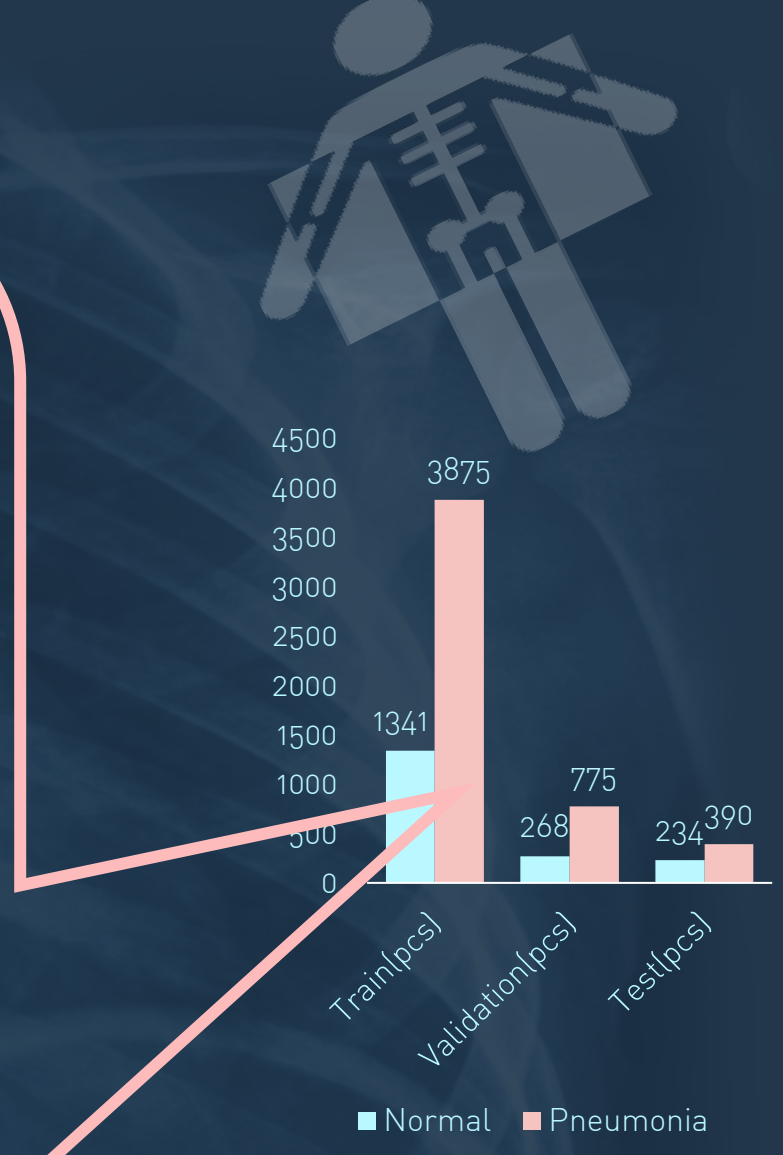
Becteria

Virus

Normal!

Data Source - from kaggle





Preprocessing

- rotation_range=15 ~ 30
- width_shift_range = 0.05~0.2
- height_shift_range = 0.05~0.2
- zoom_range = 0.1 ~ 0.4
- horizontal_flip = false

Training Models

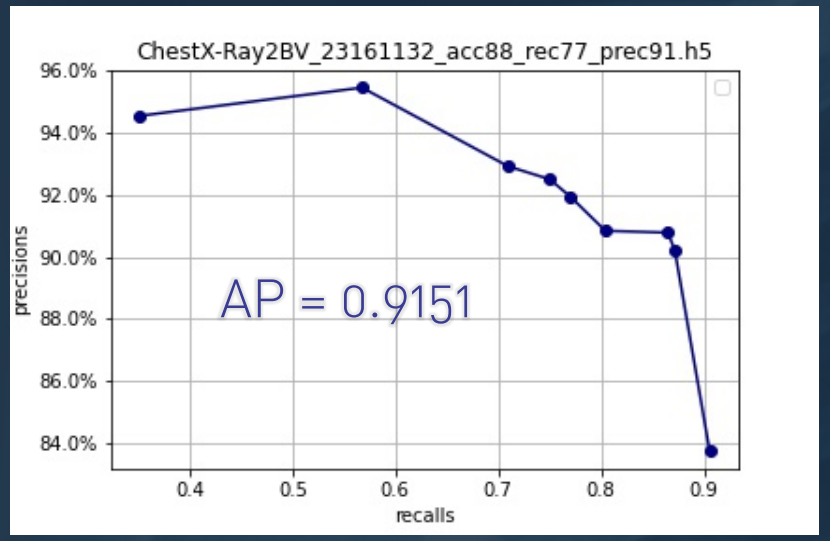
- Weight = {0:1, 1:2}
- Input_shape = 224~299
- Epoch = 30~45
- Batch size = 32~64
- Model_monitor = accuracy
- EarlyStopping
  - monitor = loss
- ReduceLROnPlateau
  - factor = 0.2~0.3
  - min_delta = 1e-3~1e-5

Criteria

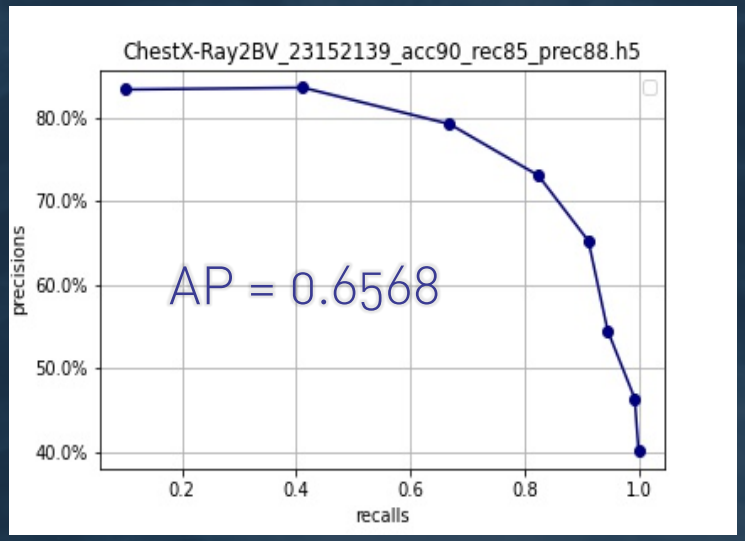
Prediction \ Actual	True	False
True	TP	FP
False	FN	TN

- Accuracy = (TP + TN)/(TP + FP + FN + TN)
- Precision = TP / (TP + FP)
- Recall= TP / (TP + FN)
- F1 score = 2 / (Precision^{-1} + Recall^{-1})

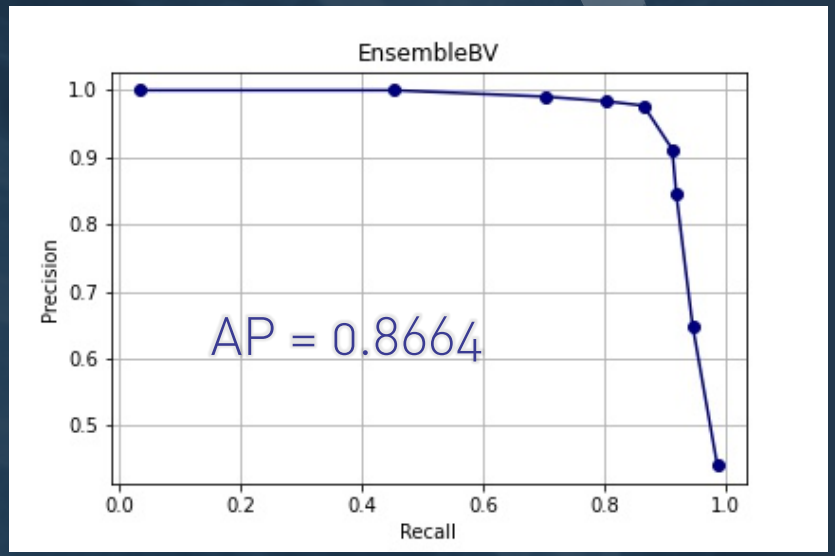
Comparison – PR curve



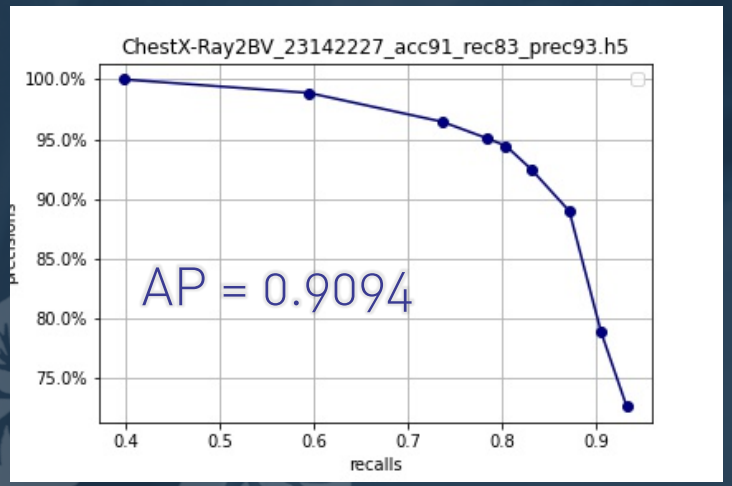
Mobilenet



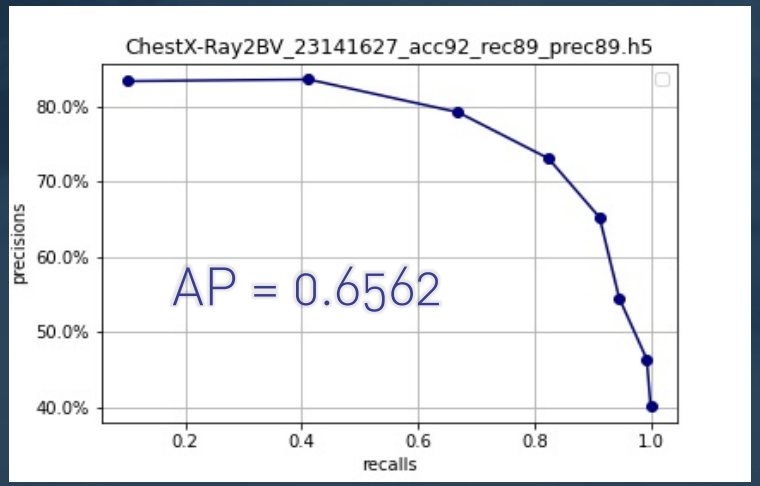
ResNet50



Ensembled



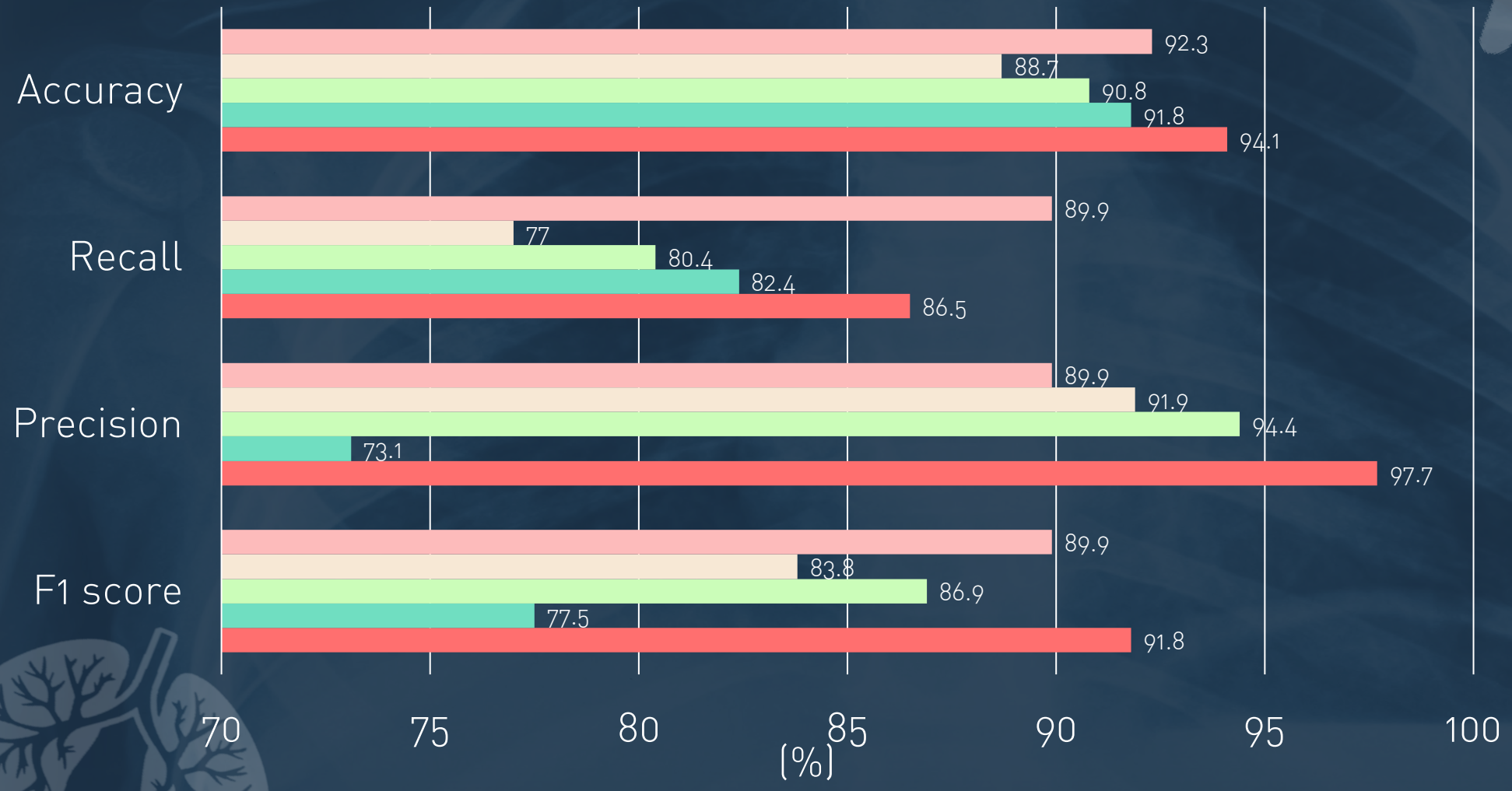
ResNet152V2



Self-build CNN

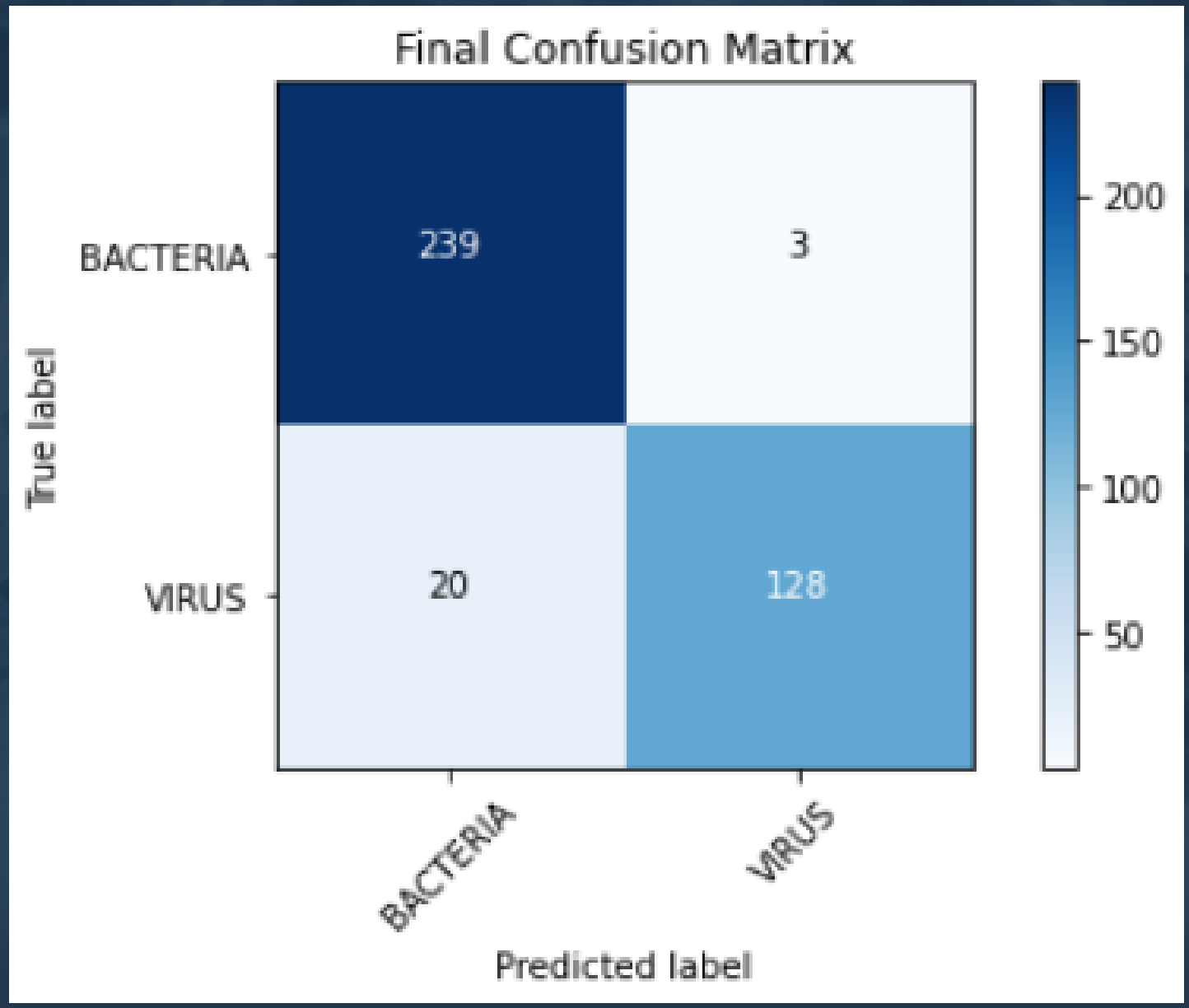
Result

Becteria / Virus

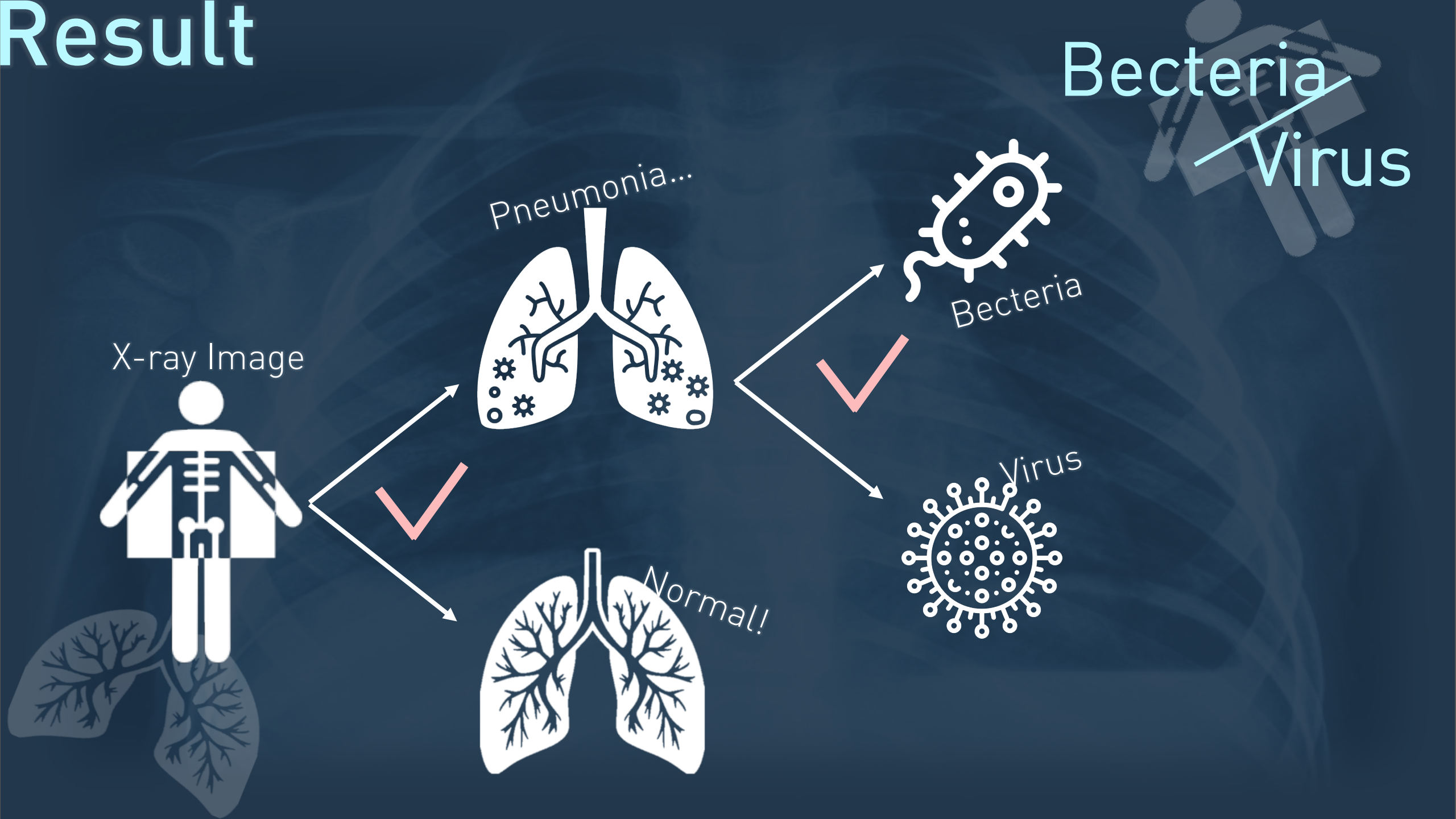


Result

Becteria
Virus



Result


Becteria
Virus
Pneumonia...
Becteria
X-ray Image
Virus
Normal!

Future Outlook

Process	Present Problem	The Way to Improve
Data Source	Resolution of X-ray imaging affects the accuracy of prediction.	Choose clear X-ray images.
Classification	Only divided into "Virus" & "Bacteria" when pneumonia is predicted.	Classify specific pathogens (e.g. tuberculosis) or virus type
Prediction	Unable to handle the coexistence of viruses and bacteria.	Use Normal / Becteria & Normal / Virus to classify.
Ensembling	Extreme results affect the average.	Choose "vote" as the ensembling way but not "average".
Competitiveness	No other ready-made model to compare.	Compare with the result of the one generated by Teachable Machine.

Future Outlook

Reference

- ✓ https=//www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia/code
- ✓ https=//www.kaggle.com/code/karnikakapoor/pneumonia-diagnosis-convnet-model
- ✓ https=//www.kaggle.com/code/madz2000/pneumonia-detection-using-cnn-92-6-accuracy
- ✓ https=//www.kaggle.com/code/alisultanov/pneumonia-diagnosis-using-resne
- ✓ https=//www.kaggle.com/competitions/vinbigdata-chest-xray-abnormalities-detection/data
- ✓ https=//medlineplus.gov/xrays.html
- ✓ https://time.com/6246045/collapse-us-health-care-system/
- ✓ https://www.scielo.br/j/rsbmt/a/98LMbshKyrXVc7sC4nZwkSG/?lang=en
- ✓ https://finance.yahoo.com/news/global-health-care-collapse-190439531.html