Recitation 7
MLHC 4/7/2023

Hussein Mozannar

The State of the Course

- Problem set 4 due in two weeks and so
- Project checkpoint 2 due next week

What's left:
➤ Final project report
➤ Project poster session
➤ Final exam (we will have prep in recitation)

Today's Recitation

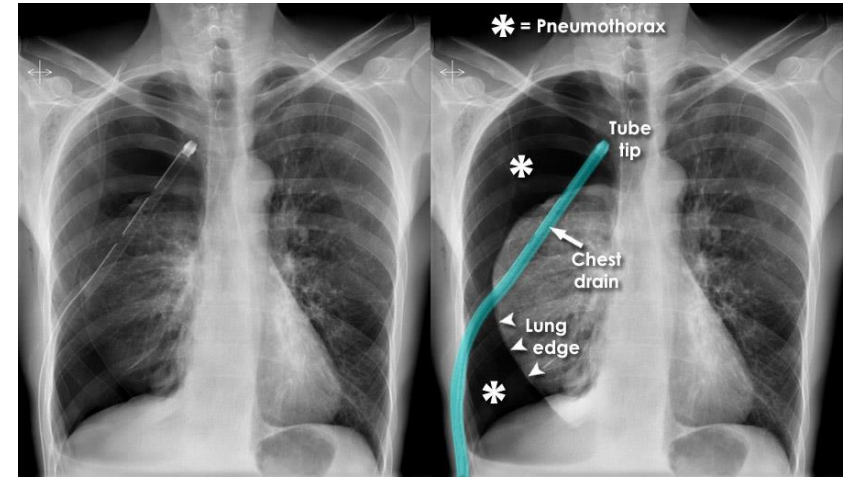Cover needed background for problem set 4

- Pneumothorax: who's a good radiologist?

- Training an AI radiologist

- Looking into the AI's brain (well not really, some interpretability)

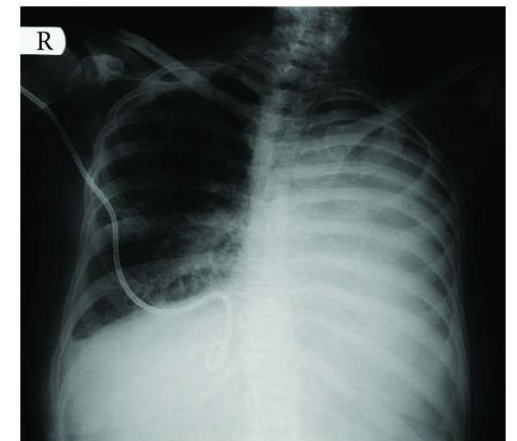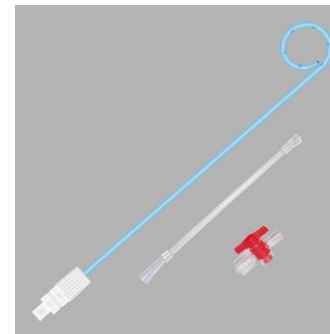- Code walkthrough

- Radiology Reports

Pneumothorax

- **What is it:** is a collapsed lung and occurs when air leaks into the space between your lung and chest wall. This air pushes on the outside of your lung and makes it collapse.
- **Causes:** blunt or penetrating chest injury, certain medical procedures, or damage from underlying lung disease
- **Symptoms:** sudden chest pain and shortness of breath, can be life-threatening

Pneumothorax Treatments

- chest tube placed between the ribs into the space around the lungs helps drain the air and allows the lung to re-expand.

- pigtail catheters

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8975284/
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3015347/

Pneumothorax: Yes or No?

Pneumothorax: Yes or No?

Pneumothorax: Yes or No?

Pneumothorax: Yes or No?

MIMIC-CXR

EXAMINATION: CHEST (PA AND LAT)

INDICATION: ____ year old woman with ?pleural effusion // ?pleural effusion
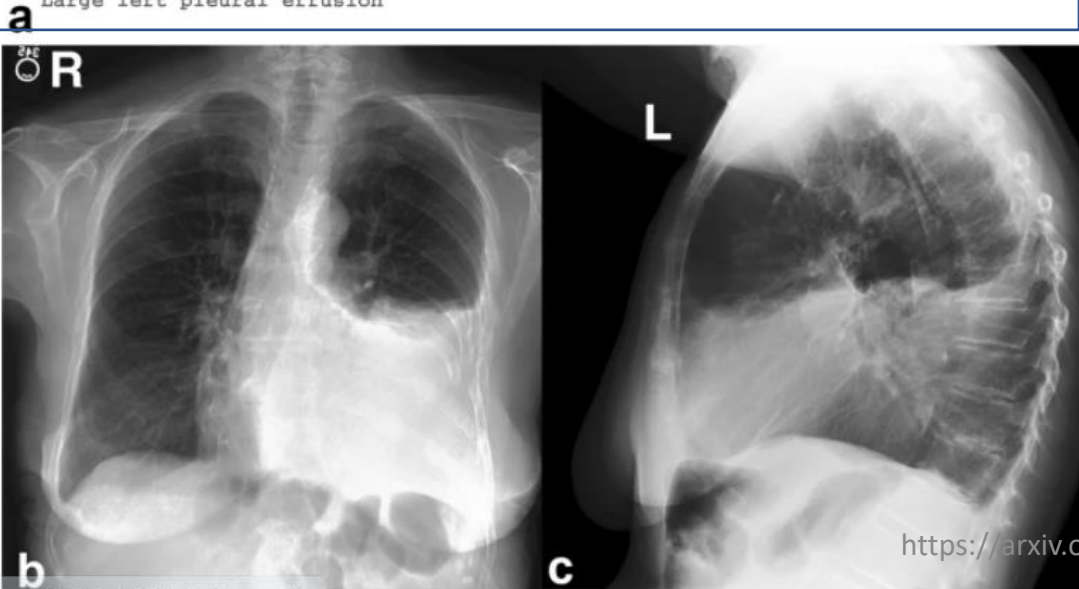
TECHNIQUE: Chest PA and lateral

COMPARISON: ____

FINDINGS:

Cardiac size cannot be evaluated. Large left pleural effusion is new.   Small right effusion is new.   The upper lungs are clear.   Right lower lobe opacities are better seen in prior CT.   There is no pneumothorax.   There are mild degenerative changes in the thoracic spine

IMPRESSION:

Large left pleural effusion

a

PLEURAL EFFUSION

The label pleural effusion is obtained using CheXpert which is a rule based classifier

MIMIC-CXR: label distribution

Table 2: Frequency of labels in MIMIC-CXR-JPG on the training subset of 222,750 unique radiologic studies (8 studies were not labeled).

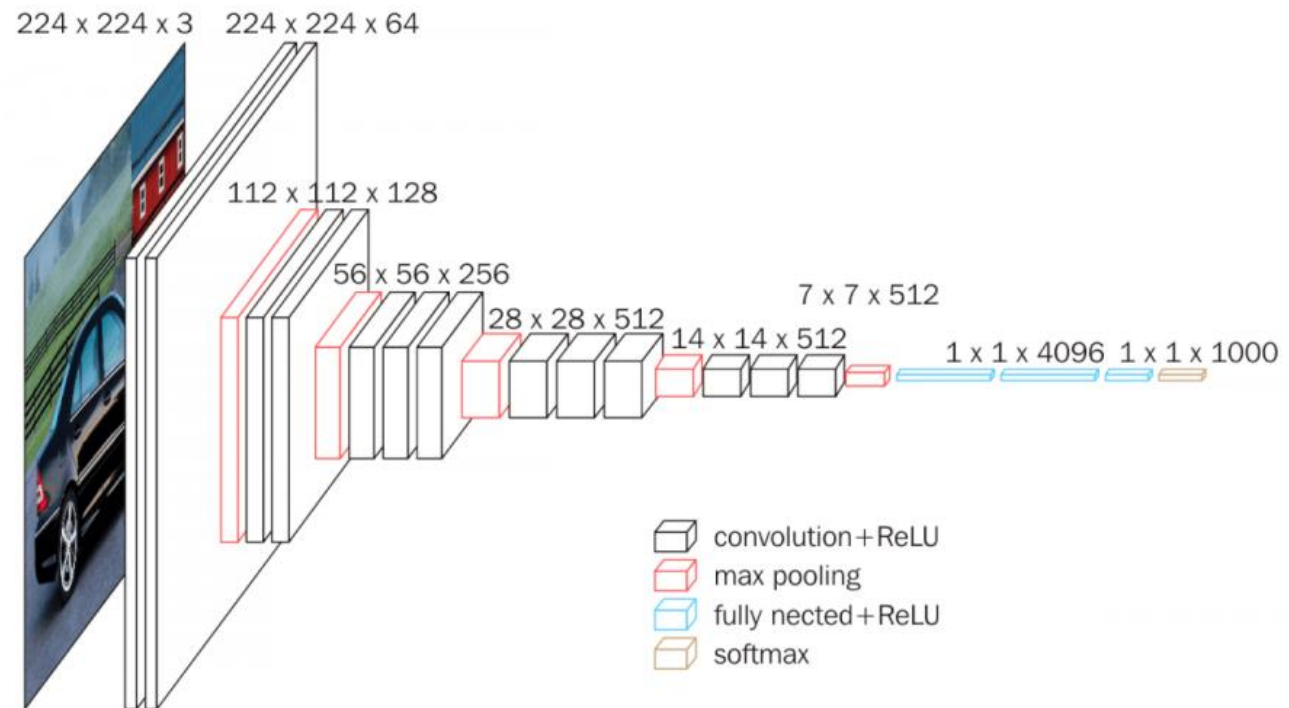|	Positive	Negative	Uncertain	Disagreement
Atelectasis	45,088 (19.8%)	937.0 (0.4%)	9,897.0 (4.3%)	1,744 (0.8%)
Cardiomegaly	39,094 (17.2%)	15,860.0 (7.0%)	5,924.0 (2.6%)	5,924 (2.6%)
Consolidation	10,487 (4.6%)	7,939.0 (3.5%)	3,022.0 (1.3%)	1,628 (0.7%)
Edema	26,455 (11.6%)	25,246.0 (11.1%)	11,781.0 (5.2%)	2,351 (1.0%)
Enlarged Cardiomediastinum	7,004 (3.1%)	5,271.0 (2.3%)	9,307.0 (4.1%)	255 (0.1%)
Fracture	3,768 (1.7%)	880.0 (0.4%)	299.0 (0.1%)	884 (0.4%)
Lung Lesion	6,129 (2.7%)	842.0 (0.4%)	1,020.0 (0.4%)	296 (0.1%)
Lung Opacity	50,916 (22.3%)	2,868.0 (1.3%)	2,110.0 (0.9%)	2,531 (1.1%)
No Finding	75,163 (33.0%)	-	-	3,906 (1.7%)
Pleural Effusion	53,188 (23.3%)	27,072.0 (11.9%)	5,345.0 (2.3%)	1,667 (0.7%)
Pleural Other	1,961 (0.9%)	120.0 (0.1%)	728.0 (0.3%)	93 (0.0%)
Pneumonia	15,769 (6.9%)	24,205.0 (10.6%)	17,789.0 (7.8%)	1,422 (0.6%)
Pneumothorax	9,317 (4.1%)	42,335.0 (18.6%)	868.0 (0.4%)	1,328 (0.6%)
Support Devices	65,637 (28.8%)	3,070.0 (1.3%)	96.0 (0.0%)	1,831 (0.8%)

MIMIC-CXR: Label validation

Table 4: Evaluation of the CheXpert mention patterns on 687 manually labeled reports. The aim is to detect any utterance of the finding, regardless of uncertainty. The CheXpert mention patterns were used for both NegBio and CheXpert in this work.

Finding	Precision	Recall	F1	Number of positive cases
No Finding	0.382	0.867	0.531	30
Enlarged Cardiomediastinum	0.375	0.600	0.462	70
Cardiomegaly	0.814	0.910	0.859	235
Lung Lesion	0.861	0.848	0.855	66
Lung Opacity	0.715	0.907	0.800	194
Edema	0.799	1.000	0.888	227
Consolidation	0.886	0.979	0.930	95
Pneumonia	0.928	0.986	0.956	223
Atelectasis	0.893	1.000	0.944	218
Pneumothorax	0.945	0.995	0.970	226
Pleural Effusion	0.968	0.978	0.973	370
Pleural Other	0.477	0.778	0.592	27
Fracture	0.870	0.940	0.904	50
Support Devices	0.823	0.974	0.892	234

Model: VGG 16

- A relatively old convolutional neural network (16 layers)

224 x 224 x 3   224 x 224 x 64

112 x 112 x 128

56 x 56 x 256

28 x 28 x 512

14 x 14 x 512

7 x 7 x 512

1 x 1 x 4096   1 x 1 x 1000

convolution+ReLU
max pooling
fully nected+ReLU
softmax

https://arxiv.org/abs/1409.1556

Loss function: Multi-Task Learning

- Instead of learning 14 different models for each condition, we train a single model that can do 14 tasks!

- Last model layer is of size 14 (one unit for each binary task)

- Loss function then becomes the sum of 14 cross entropy losses!

Data Transformations

- Data augmentation is crucial to get good results!
- How it works:
  - Each time we load a batch of data, we apply the transforms (randomly) to each image independently
  - The label is not transformed for the image
  - Train on the batch
- Since we train for many epochs, we will see each image with different transforms (kind of like having more images effectively!)
- At test time, we remove all transforms

```python
transform_train = transforms.Compose([
        transforms.RandomResizedCrop(imgtransCrop),
        transforms.RandomHorizontalFlip(),
        transforms.RandomRotation(15),
        transforms.ToTensor(),
        normalize
])
```

```python
transform_test = transforms.Compose([transforms.Resize((imgtransCrop,imgtransCrop)),
                transforms.ToTensor(),
                normalize])
```

Saliency Map

- Saliency map: Which pixels contributed to the model having the predicted label and score for this particular image

Class wise Saliency map

- Given image **I**, model assigns a score $S_c(\mathbf{I})$ for making prediction of class c (this is the score before softmax/sigmoid)

- Saliency map tries to find the proto-typical image **I** that causes the model to predict as high of a score for class c

$$\arg\max_I\ S_c(I)$$

- Since the image can be unbounded we add regularization:

$$\arg\max_I\ S_c(I) - \lambda\|I\|^2_2$$

Class wise Saliency Map

- How do we solve this seemingly difficult problem!
- Suppose our model is linear:

$$S_c(I) = w_c^T I + b_c,$$

- Best image **I** can be found easily :

$$Objective = w_c^\top I + b_c - \lambda \|I\|_2^2$$

$$\text{differentiate and set to 0: } w_c - 2\lambda I = 0$$

$$\rightarrow I = w_c/(2\lambda)$$

- (set lambda to ½ ☺) Our CNN is not linear though…

From class-wise to instance specific

- In the example in the previous slides, I started with a specific image I_0

- Idea: define the **I** that maximizes the class score as a perturbation of image I_0

- Define w as the derivative of the score function with respect to the original image (just take backpropagate through the model on the image)

$$w = \left. \frac{\partial S_c}{\partial I} \right|_{I_0}.$$

- Then via Taylor expansion our scoring function can be approximated as

$$S_c(I) \approx w^T I + b$$