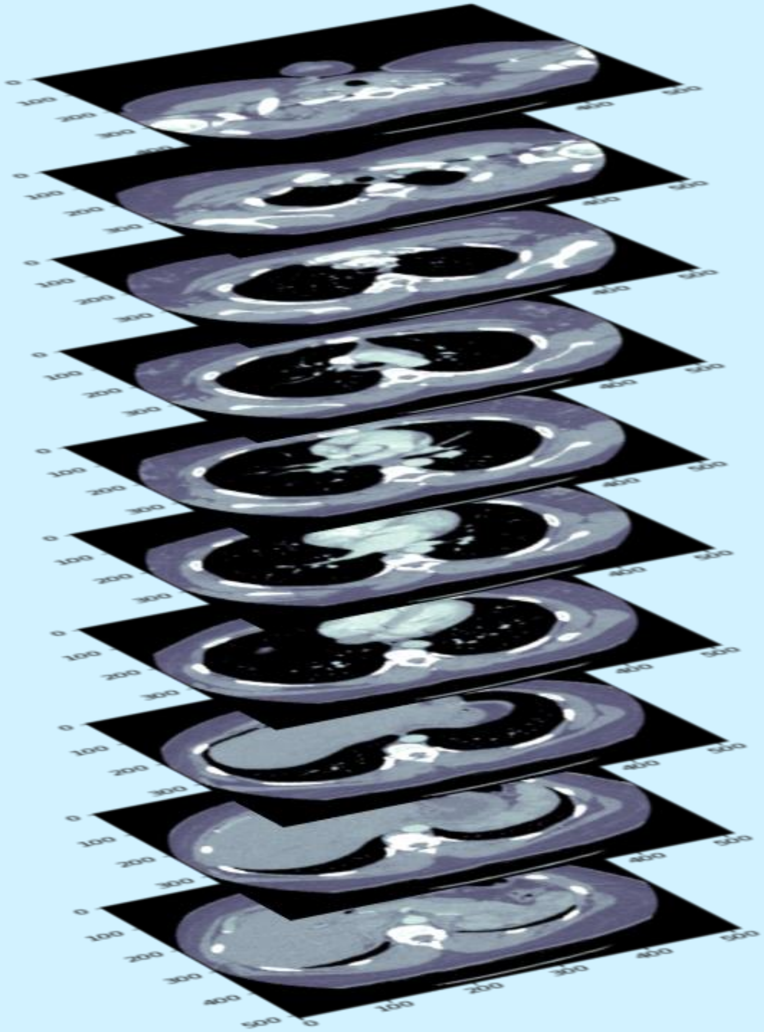
RSNA STR Pulmonary Embolism Detection

7th Place

Yuval Reina



Agenda

1. Background
2. Summary
3. Base Models
4. Transformer Models
5. 2^{nd} Opinion Ensembling
6. Next step

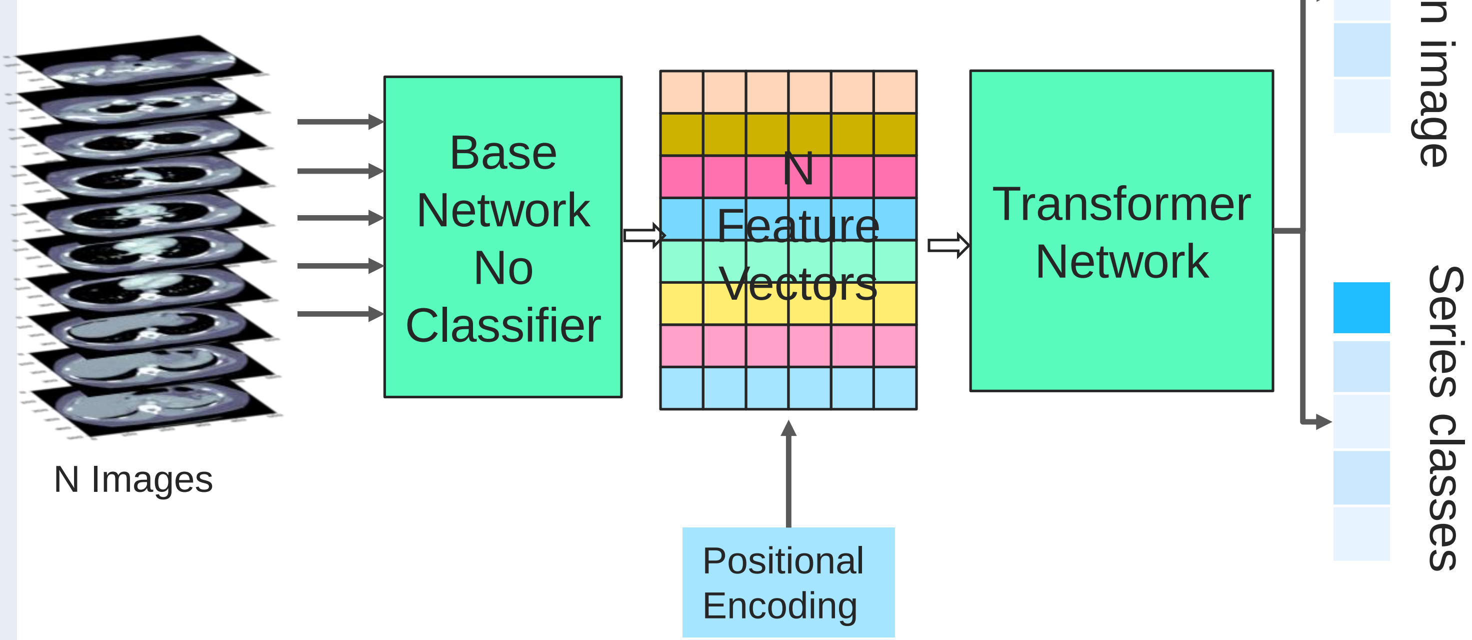
Background

- Yuval Reina
  - BScEE, MBA, VP R&D @ D-Fend
  - Hobbyist, Self-Education

Summary

Two stage solution:

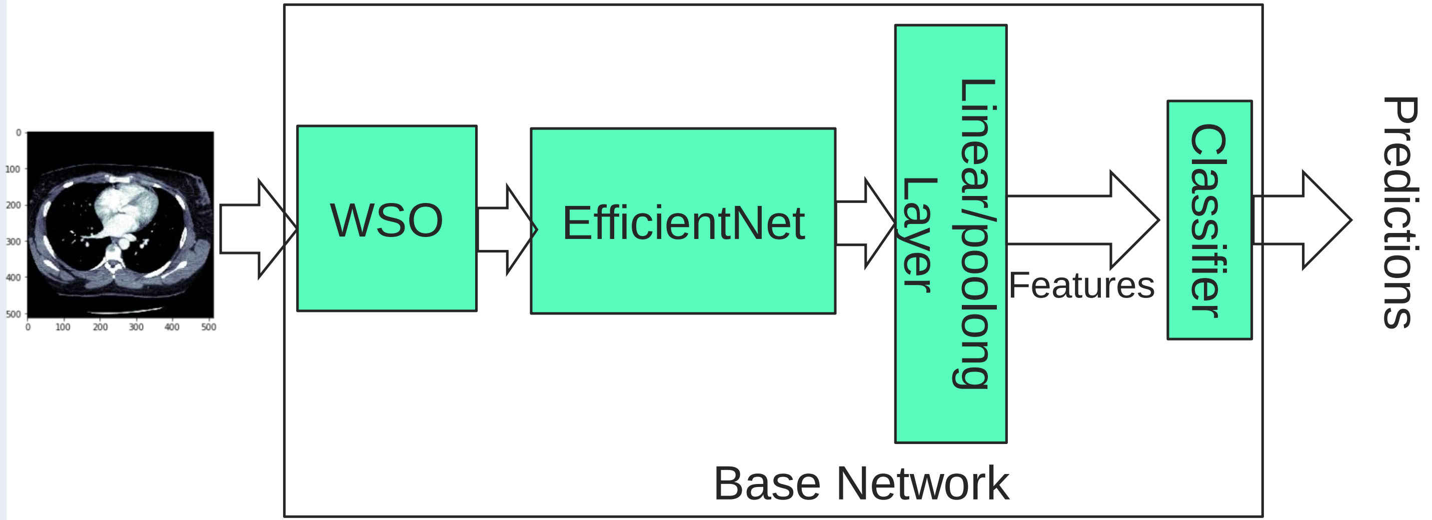
- Base model for feature extraction per image
- Transformer model for context learning



Summary

Two stage solution:

- Base model for feature extraction per image



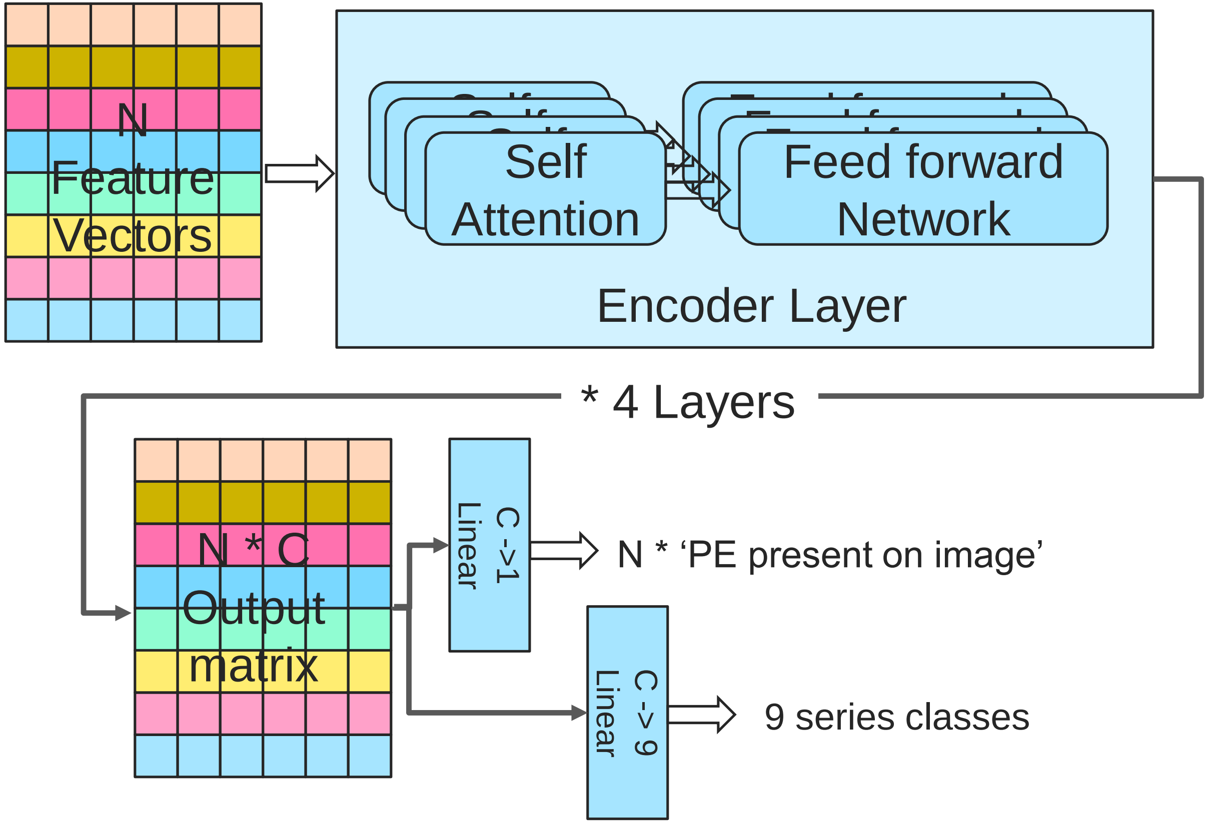
Pre – trained models:

- EfficientNet B3, B5

Summary

Two stage solution:

- Transformer model for context learning



Base Models

As base model I used different types of EfficientNet
B3, B5
Pre-trained on Imagenet using Noisy student algorithm

Loss Functions:

- Weighted BCE on ‘PE present on image’ and the series classes
- If ‘PE present on image’ =0 then all series classes are set to 0, except the QA classes

Number of Features – 256

WSO - Windowing

The dynamic range of the CT pixel values is very large.

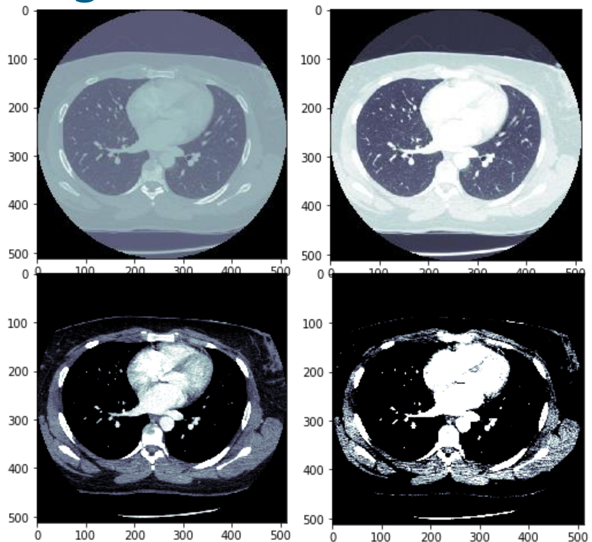
One way to handle this is with windowing as seen below.

In our model we used adaptive windowing:

$P_{new} = \text{sigmoid}(a * P_{orig} + b)$

a, b are optimized by the network

This is easily achieved by using a 2D conv with 1*1 kernel.



This idea was introduced In:

Hyunkwang Lee, Myeongchan Kim, Synho Do, Practical Window Setting Optimization for Medical Image Deep Learning, https://arxiv.org/pdf/1812.00572.pdf

Base Models – Cont.

Augmentation

- Random resize + crop
- Random rotation
- Random flip
- Random shift of CT's of mean and std
- Cutout - erasing a small rectangle in the image

Training

- Use sampler to emphasize the positive targets and slices near the center
- Epoch size = 0.15*training data

Feature extraction

The feature where extracted by the output of the last layer before the classification layer.

Transformer Model

- The transformer is a stack of 4 - 6 transformer encoder as described in Attention Is All You Need with 2 - 4 attention heads
- The input is N feature vectors from the same series.
- N = 128, if the patient has more then 127 images, they are randomly grouped to groups of equal size and padded to 128.
- Slice order and relative position is embedded and added to the feature vectors

Transformer Model - Output

- The output of the plain transformer is N * C matrix
- To get the models output I use 2 extra linear layers:
    - C -> 1 layer to get N * ‘PE present on Image’
    - C -> 9 layer to get 9 series classes (Run only on the N’th output vector)

Transformer Training

Augmentation

- Add noise to the feature vectors
- Different random grouping every epoch

Loss Functions:

- Mimics the competition’s metric

Inference

1. Use the base model to extract features for each CT with no augmentation
2. Randomly Divide the slices of the series to equal size groups, up to 127 and pad to 128
3. Inference using Transformer network.

TTA – repeat steps 2-3 12 times and average the results

Training And Inference Times

Setup:

- CPU – Intel i9-9920
- RAM – 64G
- GPU – Tesla V100 32G / Titan RTX (20% slower)

Training:

- Base Models: ~ 9 (B5) h/model
- Feature extraction: 3 – h/model
- Transformer: 15min/model
- Total time for all models (4) ~ 48H for 1 GPU

Inference:

- Inference 4H/model (Kaggle kernel)

Ensembling – 2nd Opition Method

Motivation : Each model's full Inference time ~ 4H on Kaggle's kernel => only 2 models can be ensembled in 9H

Solution: Do Ensembling only for series with the highest uncertainty in classification.

The uncertainty is high if the classification value is near 0.5

The procedure

- full inference with 1st model.
- use the uncertainty to select about 50% of the series and interface with a 2nd model and Ensemble
- use the uncertainty to select about 50% of the series and interface with a 3rd model and Ensemble
- use the uncertainty to select about 25% of the series and interface with a 4th model and Ensemble

Ensembling – Models Used

The models which were used in the final solution are:

1. base - EfficientNet b5 noisy student, fold 0/5, transformer - 4 encoders, 2 attention heads, 2048 feedforward dim.
2. base - EfficientNet b5 noisy student, fold 2/5, transformer - 4 encoders, 2 attention heads, 3072 feedforward dim.
3. base - EfficientNet b5 noisy student, fold 1/5, transformer - 4 encoders, 4 attention heads, 3072 feedforward dim.
4. base - EfficientNet b3 noisy student, fold 0/5, transformer - 6 encoders, 4 attention heads, 2048 feedforward dim.

Results

Private LB - 0.157

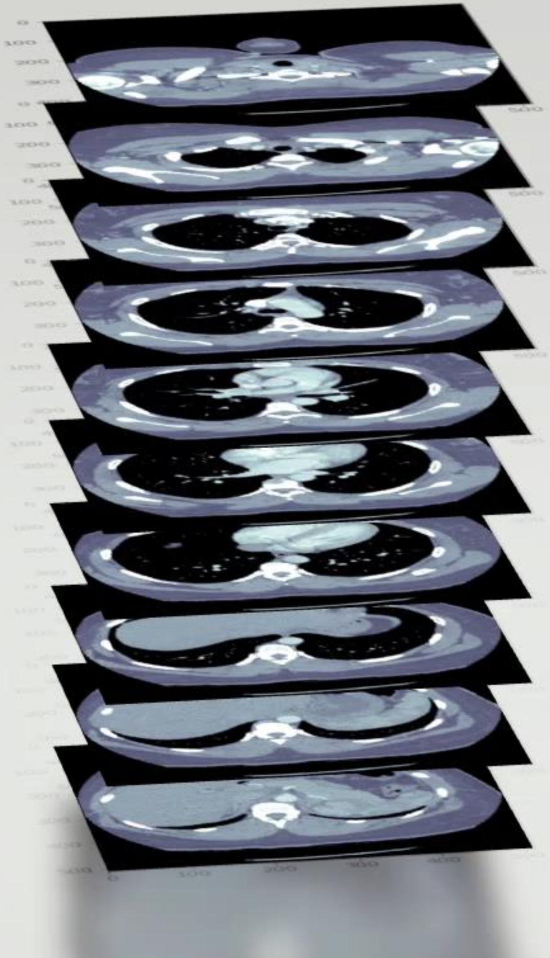
Public LB - 0.158

Next steps

Unified model for CT scans

- The current model (without ensembling) is quit simple and flexible. This type of model could have scored high also in 2019 ‘RSNA Intracranial Hemorrhage Detection’
- The heavy lifting in the model is the base model training
- A base model can be trained on various types of CT scans (Lungs, Head, etc.)
- This base model can be used without fine-tuning to extract features for various tasks related to CT scans.
- The only fine-tuning will be done on the 2^{nd} layer (transformer, or other type of layer)

RSNA



Kaggle