SIIM-ACR Pneumothorax Segmentation

1~10등 솔루션

1. Overview

(1) Overview

Overview

문제
- 흉부 방사선 사진에서 기흉을 분류

평가
- Dice Coefficient

제출
- Run-length encoding

데이터
- 12047 학습 이미지

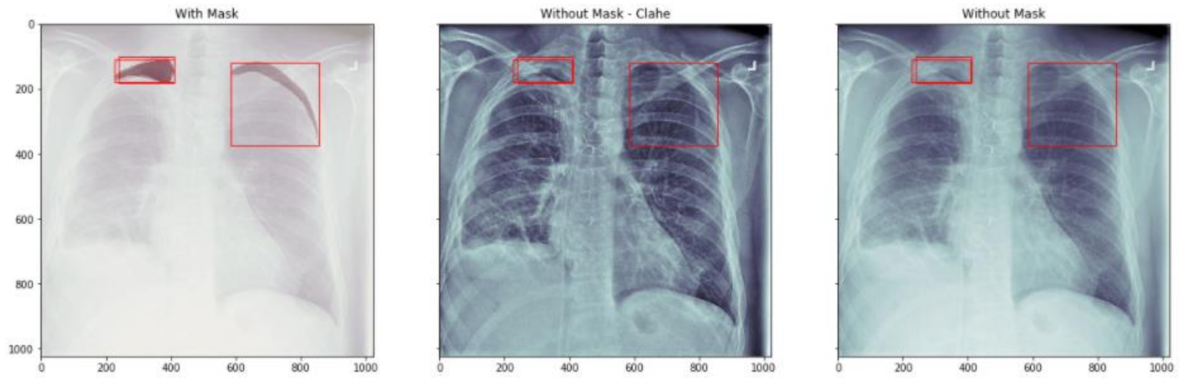
출처 : https://www.kaggle.com/jesperdramsch/intro-chest-xray-dicom-viz-u-nets-full-data

1. Overview

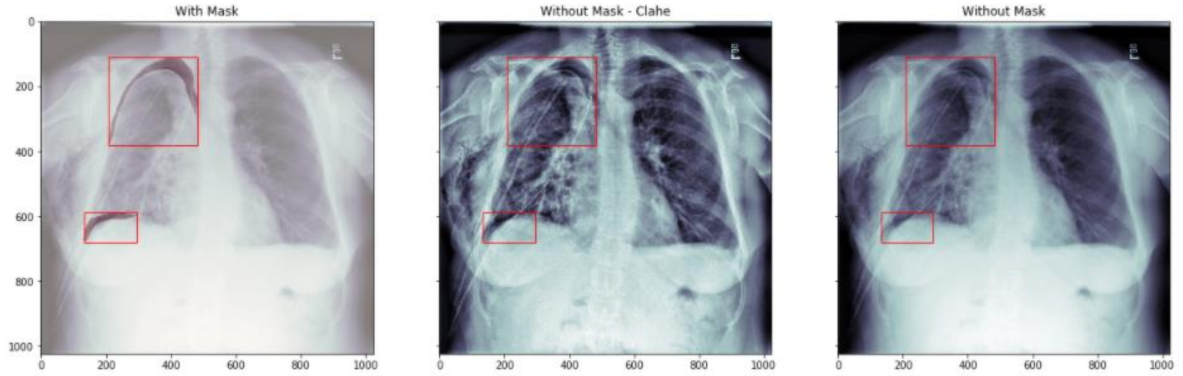
(1) Overview

Overview

```
image id: 1.2.276.0.7230010.3.1.4.8323329.406.1517875162.818172
```




```
image id: 1.2.276.0.7230010.3.1.4.8323329.3965.1517875180.406042
```



출처 : https://www.kaggle.com/ekhtiar/finding-pneumo-part-1-eda-and-unet

2. Solution
(1) 1st Solution

Solution

1st solution

Network
- AlbuNet (resnet 34) from TernausNet
- ResNet50 from selim_sef SpaceNet 4
- scSEUnet (seresnext50) from selim_sef SpaceNet 4

Loss
- Combinations of BCE, dice and focal loss
- 각각 모델에 대해 best experiment weight 찾아 사용

Augmentation
- Horizontal Flip, Random Contrast, Random Brightness, ElasticTransform, Grid Distortion, Optical Distortion, ShiftScaleRotate (Albumentations)

Code
- https://github.com/sneddy/kaggle-pneumathorax

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107824

Solution

1st solution

Learning process

- Sliding sample rate & learning rate

  : 학습 초반엔 방향성을 빠르게 잡을 수 있게, 학습 후반에는 섬세하게!

  Part 0 – pretrained model with large lr (1e-3 or 1e-4), large sample rate (0.8)

  Part 1 – uptrain the best model from previous step with normal lr (~1e-5), large sample rate (0.6)

  Part 2 – uptrain the best model from previous step with normal lr, small sample rate (0.4)

  Part 3 – simple uptrain with relatively small lr(1e-5 or 1e-6), small sample rate (0.6)

- Uptrain from lower resolution (512x512 → 1024x1024)

- Small batch size(2~4) without accumulation

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107824

2. Solution
(1) 1^{st} Solution

Solution

1^{st} solution

Other [0.75, 2000, 0.3]

- Triplet scheme (top_score_threshold, min_contour_area, bottom_score_threshold)

```
classification_mask = predicted > top_score_threshold
mask = predicted.copy()
mask[classification_mask.sum(axis=(1,2,3)) < min_contour_area, :,:,:] =
np.zeros_like(predicted[0])
mask = mask > bot_score_threshold
return mask
```

- top_score_threshold is simple binarization threshold and transform basic sigmoid mask into a discrete mask of zeros and ones.
- min_contour_area is the maximum allowed number of pixels with a value greater than top_score_threshold

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107824

Solution

1st solution

Other

- Horizontal Flip TTA
- Checkpoints averaging

: Top 3 checkpoints averaging from each fold from each pipeline on inference

Experiment name	Train info	Fold 0	Fold 1	Fold 2	Fold 3	Fold 4	Mean by folds	Submit Info	Public LB	Exist Pneumothorax
AlbunetPublic-512	TRAIN_TRANSFORMS: transforms/train_transforms_complex_512_old.json VALID_TRANSFORMS: transforms/valid_transforms_512_old.json	0.8528	0.8503	0.8499	0.8533	0.8493	**0.85112**	best 'AREA_THRESHOLD': 1000, 'SCORE_THRESHOLD': 0.3,	0.8753	221
								top3 'AREA_THRESHOLD': 1000, 'SCORE_THRESHOLD': 0.3,	0.8757	231
AlbunetPublic	TRAIN_TRANSFORMS: transforms/train_transforms_complex_1024_old.json VALID_TRANSFORMS: transforms/valid_transforms_1024_old.json	0.857	0.856	0.857	0.8539	0.85551	**0.8558**	AREA_THRESHOLD': 2250, 'SCORE_THRESHOLD': 0.45,	0.8827	257
								AREA_THRESHOLD: 600 TOP_SCORE_THRESHOLD: 0.75 BOTTOM_SCORE_THRESHOLD: 0.4	0.8871	251
SeUnet-512	TRAIN_TRANSFORMS': 'transforms/train_transforms_complex_512_old.json', 'VALID_TRANSFORMS': 'transforms/valid_transforms_512_old.json'	0.85391	0.8525	0.8521	0.85648	0.8511	**0.853218**	AREA_THRESHOLD: 600 TOP_SCORE_THRESHOLD: 0.75 BOTTOM_SCORE_THRESHOLD: 0.4	0.8777	252
SeUnet-1024	TRAIN_TRANSFORMS: transforms/train_transforms_complex_1024_old.json VALID_TRANSFORMS: transforms/valid_transforms_1024_old.json	0.8634	0.8584	0.8601	0.859	0.8611	**0.8604**	AREA_THRESHOLD: 600 TOP_SCORE_THRESHOLD: 0.75 BOTTOM_SCORE_THRESHOLD: 0.4	0.88	261
AlbunetValid-512	transforms/train_transforms_complex_512_old.json'	0.8554	0.85334	0.852	0.85431	0.8507	**0.85315**			
AlbunetValid	transforms/train_transforms_complex_1024_old.json'	0.85771	0.8535	0.8591	0.86154	0.85718	**0.857806**	AREA_THRESHOLD: 600 TOP_SCORE_THRESHOLD: 0.75 BOTTOM_SCORE_THRESHOLD: 0.4	0.8842	254
Resnet50	train_transforms_complex_1024.json	0.85951	0.85674	0.85614	0.8583	0.86039	**0.858216**	AREA_THRESHOLD: 600 TOP_SCORE_THRESHOLD: 0.75 BOTTOM_SCORE_THRESHOLD: 0.4	0.8831	228

Solution

2nd solution

Classification

- Pneumothorax or not
- BCE + Focal Loss
- Augmentation : Hflip, Scale, Rotate, Bright, Blur
- Backbone: seresnext 50, seresnext101, efficientnet-b3
- Ensemble: stacking

Solution

2nd solution

Segmentation

- Unet, DeepLabv3
- Dice Loss
- Augmentation : Hflip, Scale, Rotate, Bright, Blur
- Backbone: seresnext50, seresnext101, efficientnet-b3, efficientnet-b5
- Ensemble: average

Code

- https://github.com/yelanlan/Pneumothorax-Segmentation-2nd-place-solution

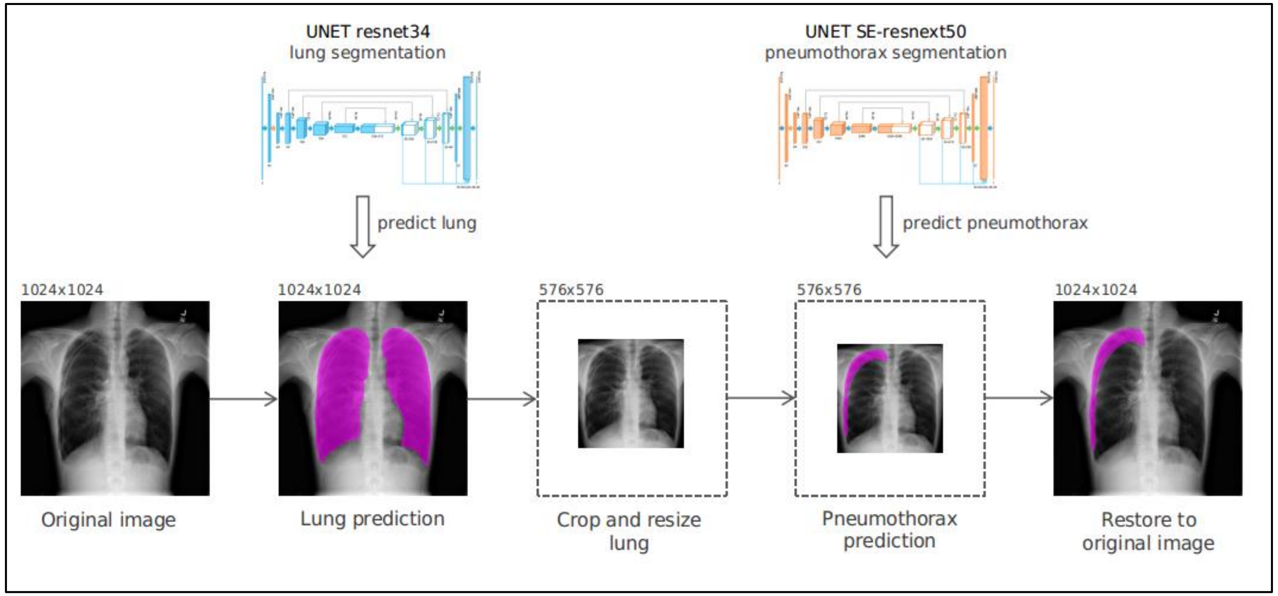
출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/108009

2. Solution

(3) 3rd Solution

Solution

3rd solution



Network

- UNet(ResNet50, SE-resnext50)

1) predict lung

2) segmentation on cropped lung

→ memory와 computational cost 아끼기 위해!

Code

- https://github.com/bestfitting/kaggle/tree/master/siim_acr

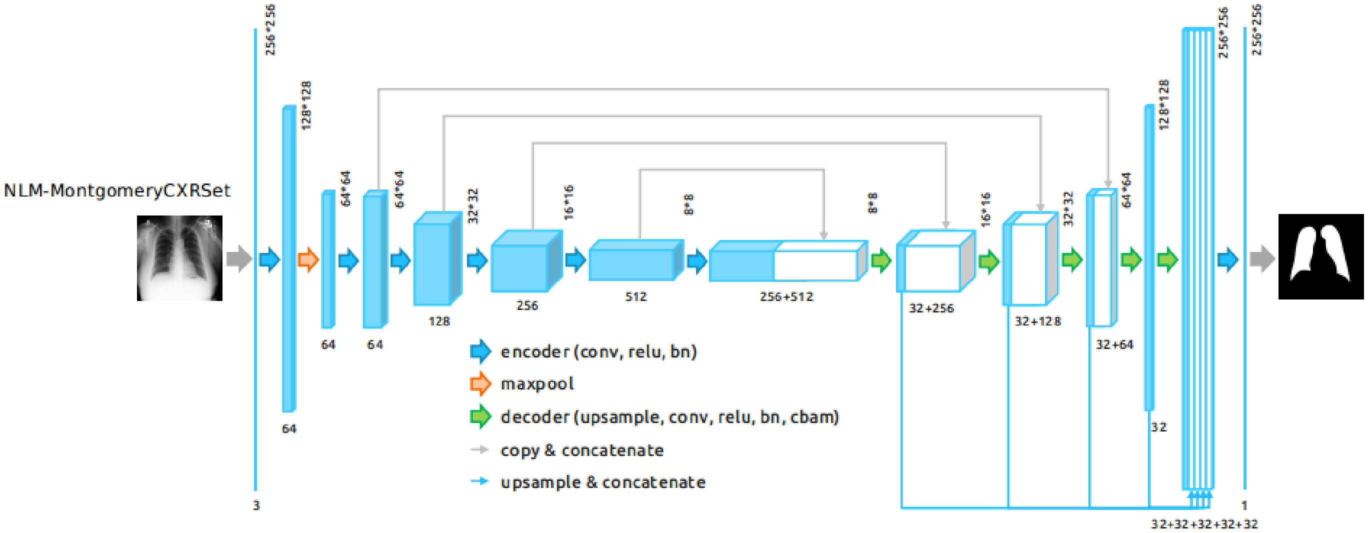
출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107981

Solution

3rd solution

Lung 부분만 Segmentation 해서 Crop Image 추출

Lung Segmentation:UNET Resnet34



Settings

Dataset: NLM-MontgomeryCXRSet,138 samples
Attention: CBAM
Loss: Symmetric Lovasz Loss
Optimizer: Adam with 0.0001

Epochs: 30
Batch Size: 16 on a GPU
No classification model,**No** classification loss
No threshold search, just used 0.5

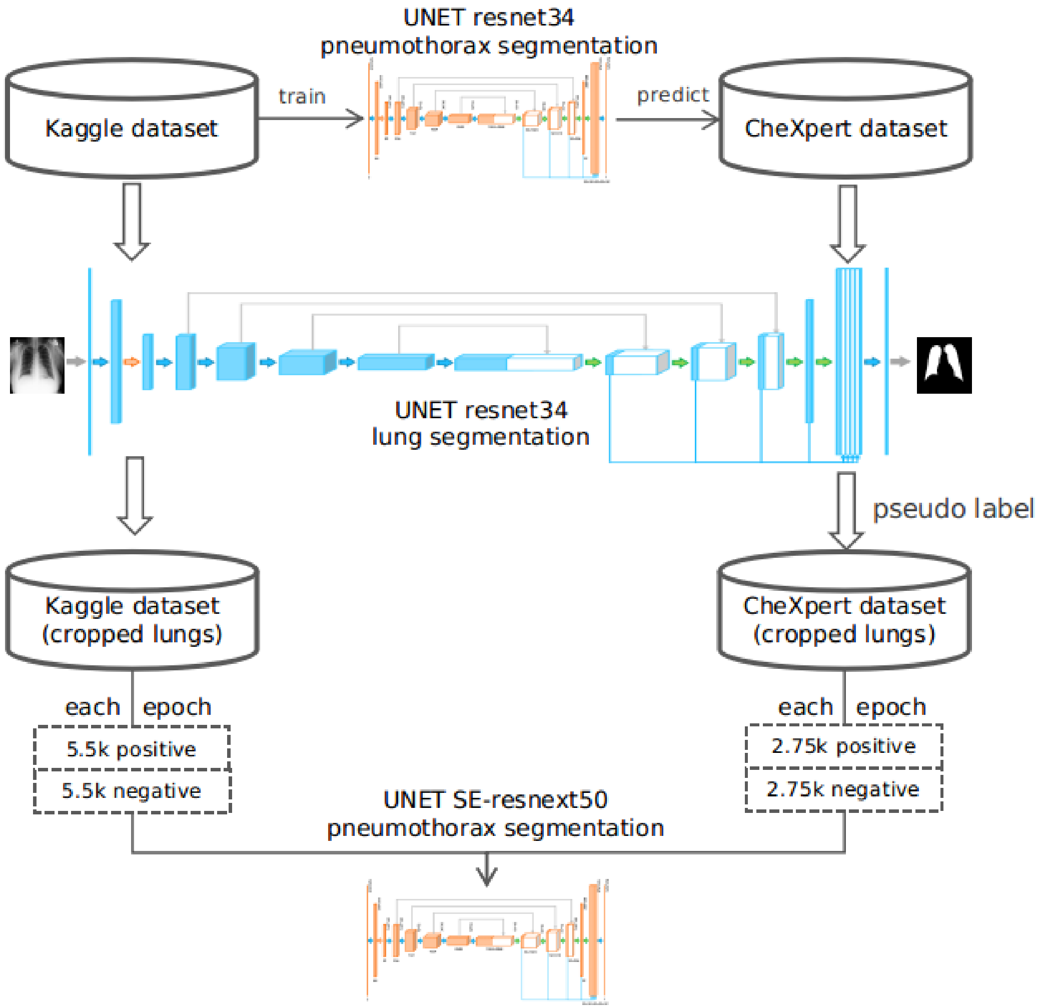
출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107981

2. Solution

(3) 3rd Solution

Solution

3rd solution



Pseudo Label을 통해서 학습 이미지 생성

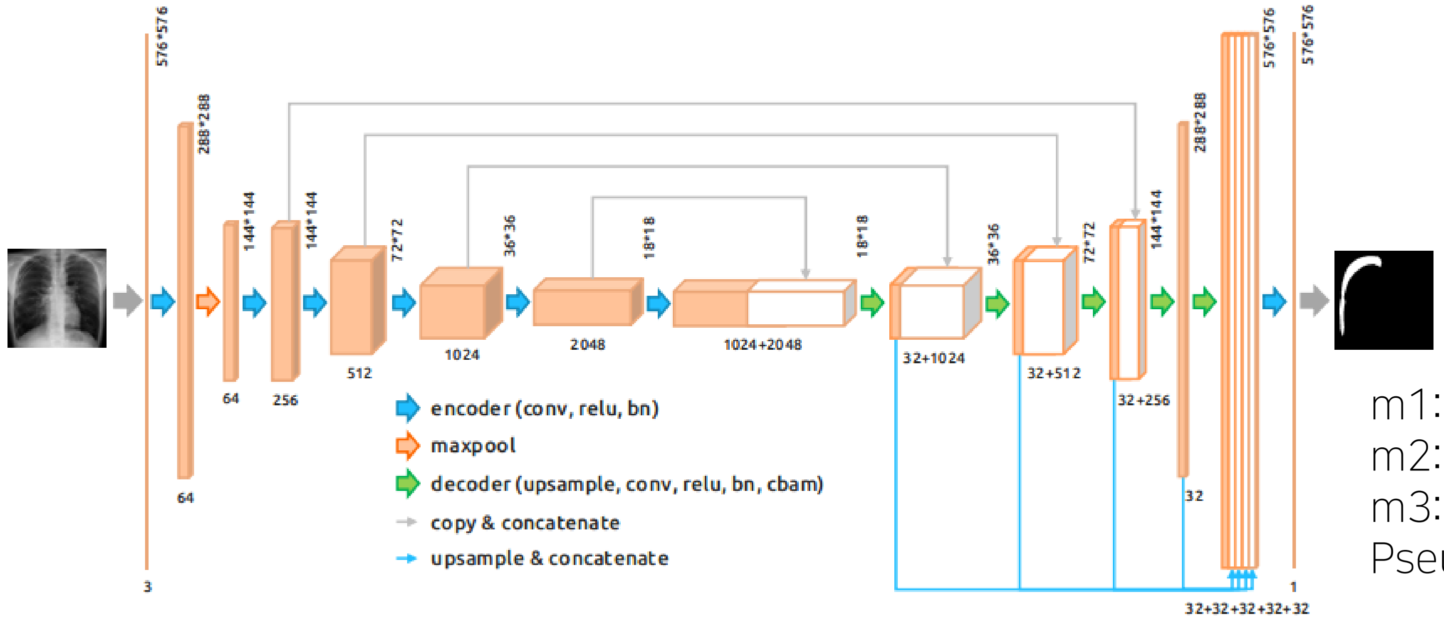
출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107981

2. Solution
(3) 3rd Solution

Solution

3rd solution

Pneumothorax Segmentation:UNET SE-Resnext50 - Public 0.8809 Private 0.8642



m1:704x704 images with no pseudo
m2:576x576 images with CheXpert Pseudo
m3:576x576 images with CheXpert and NIH Pseudo

Settings

Dataset: Kaggle dataset : CheXpert dataset = 1 : 0.5
Attention: CBAM
Loss: Lovasz Loss
Optimizer: Adam with 0.0001
Epochs: 15
EMA of model parameters

Batch Size: 3 per GPU when trained on 576x576 images,**no accumulations** of batch-size, **no Synch-BN** was used
No classification model,**No** classification loss
No threshold search, just used 0.5

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107981

Solution

3rd solution

Learning process

- 외부 데이터 사용.

: 외부 데이터의 annotation이 정확하지 않아 pseudo label 부여해 사용.
Kaggle 데이터로 학습한 모델로 라벨 예측, annotation과 모델의 예측이 모두 positive인 sample만 사용함.

Attention

- CBAM

Loss

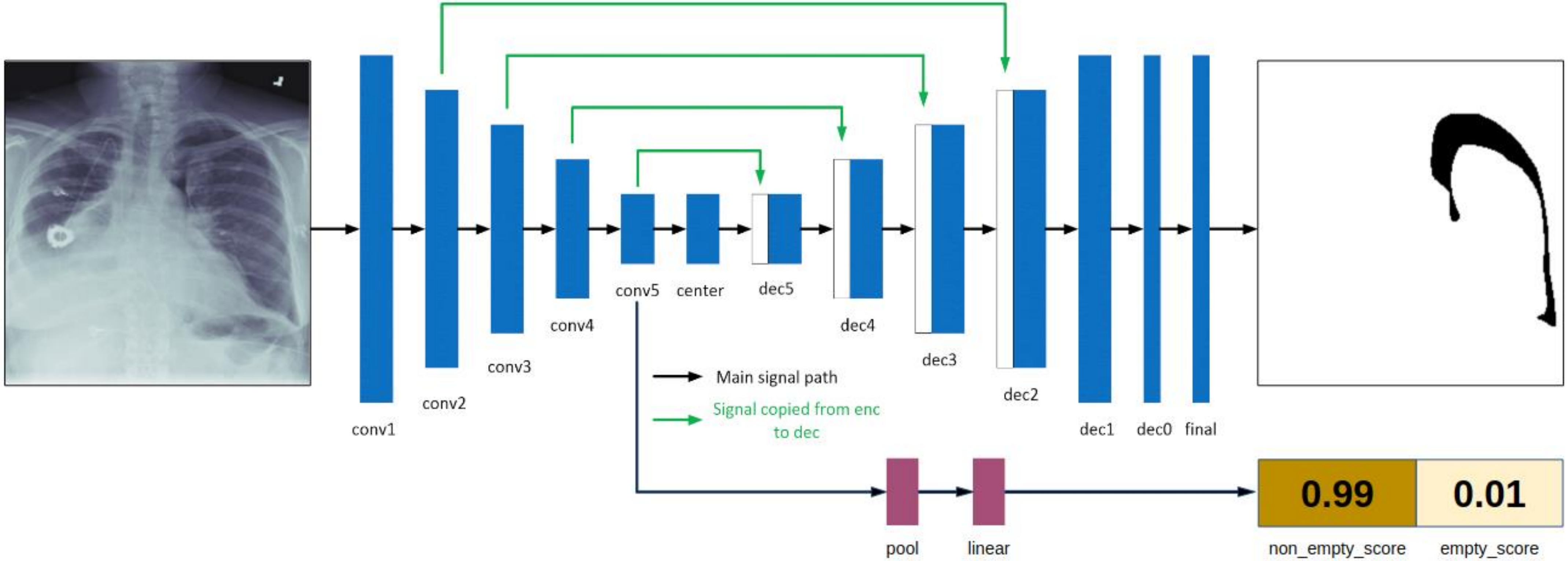
- Lovasz loss

EMA after 6 epoch

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107981

Solution

4th solution



Solution

4th solution

- Model: UNet
- Backbone: ResNet34 backbone with frozen batch-normalization
- Preprocessing: training on random crops with (512, 512) size, inference on (768, 768) size.
- Augmentations: ShiftScaleRotate, RandomBrightnessContrast, ElasticTransform, HorizontalFlip from albumentations.
- Optimizer: Adam, batch_size=8
- Scheduler: CosineAnnealingLR

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/108397

2. Solution

(4) 4th Solution

Solution

4th solution

- Additional feature: the proportion of non-empty samples linearly decreased from 0.8 to 0.22 (as in train dataset) depending on the epoch. It helped to converge faster.

- Loss: 2.7 * BCE(pred_mask, gt_mask) + 0.9 * DICE(pred_mask, gt_mask) + 0.1 * BCE(pred_empty, gt_empty). Here pred_mask is the prediction of the UNet, pred_empty is the prediction of the branch for empty mask classification.

- Postprocessing: if pred_empty > 0.4 or area(pred_mask) < 800: pred_mask = empty. Parameters are selected on the validation set.

- Ensemble: averaging the 4 best checkpoints over 8 folds, horizontal flip TTA.

Code

- https://github.com/amirassov/kaggle-pneumothorax

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/108397

Solution

5^{th} solution

Network

- Unet(SE-resnext50, SE-resnext101) with Aspp(Astrous Spatial Pyramid Pooling)

Loss

- 1024 * BCE(results, mask) + BCE(cls, cls_target)

Code

- https://github.com/earhian/SIIM-ACR-Pneumothorax-Segmentation-5th

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107603

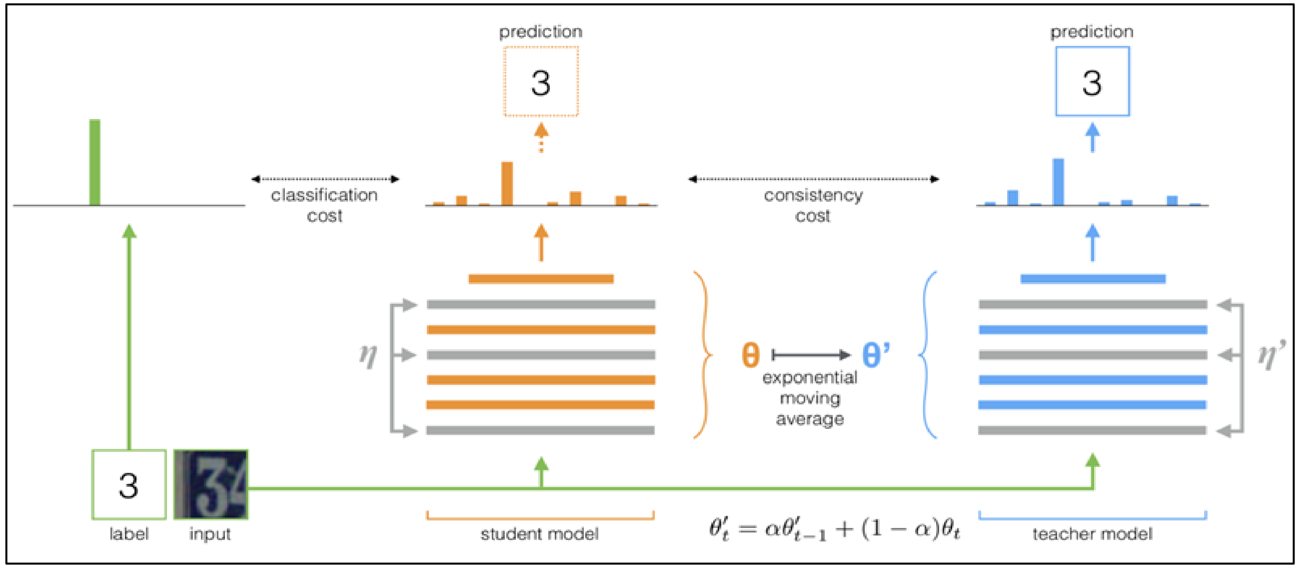
Solution

5th solution

Learning process

- Semi supervision; mean-teacher with NIH Dataset

: NIH Dataset은 100,00개의 흉부 X선 이미지로 기흉을 포함한 4개의 질병에 대한 annotation이 존재함. 이 데이터에 대해 mean-teacher라는 semi supervision 모델을 이용, train 데이터 추가함.



Solution

6th solution

Model

- EncodingNet (ResNets, 512 and 1024 size) and UNet (EfficientNet4, se-resnext50, SENet154 with 512, 640 and 1024 sizes) – trained and tuned independently

Tricks:

- classification on top of EncondingNet with heavy TTA (11 methods)
- small segments of predicted masks was deleted

Others

- Loss: BCE+Dice (tried Focal, didn't work)
- Lower image sizes reduced score a lot so full sized model should be better.

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107743

Solution

6^{th} solution

- Best augmentations were related to crops and rotations. We didn't use contrast and brightness transformations.

```
AUG = Compose(
    [
        HorizontalFlip(p=0.5),
        OneOf(
            [
                ElasticTransform(
                    alpha=300,
                    sigma=300 * 0.05,
                    alpha_affine=300 * 0.03
                ),
                GridDistortion(),
                OpticalDistortion(distort_limit=2,
shift_limit=0.5),
            ],
            p=0.3
        ),
        RandomSizedCrop(min_max_height=(900, 1024), height=1024,
width=1024, p=0.5),
        ShiftScaleRotate(rotate_limit=20, p=0.5)
    ],
    p=1
)
```

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107743

Solution

6th solution

Model1:

- EncNet (from pytorch-encoding) with Resnet50 pretrained on Pascal dataset.
- First, we trained 4-fold-blend model on 512 x 512 resolution with AUG.
- Second, we trained 4-fold-blend model on 1024 x 1024 resolution with AUG .
- Finally, we avereged two 4-fold-blend models.

- Loss: Dice
- Augmentation: AUG
- TTA: Flip, Clahe and np.arange(-20, 21, 5) angles rotations (total 10 TTA)

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107743

Solution

6^{th} solution

Model2-3:

- Two Unet-like models with SEResnext50 and SEResnet152 backbones were trained on
- 1024 x 1024 resolution. Each model also was averaged along 4 folds.

- Loss: Dice
- Augmentation: AUG
- TTA: Flip

Solution

6^{th} solution

Model4:

- Unet-like model with EfficientNetB4 backbone trained on 640 x 640 resolution.
- Also were blended along 4 folds.

- Loss: weighted Dice + BCE
- Augmentation: AUG
- TTA: no TTA

Solution

6^{th} solution

Final model:

- Average of Model1, Model2, Model3 and Model4

- Classification threshold: ~0.32
- Segmentation threshold: ~0.375

Solution

7th solution

Our solution is quite simple, it is an ensemble (simple average) of four different models (several folds each):

- FPNetResNet50 (5 folds)
- FPNetResNet101 (5 folds)
- FPNetResNet101 (7 folds with different seeds)
- PANetDilatedResNet34 (4 folds)
- PANetResNet50 (4 folds)
- EMANetResNet101 (2 folds)

Code : https://github.com/see--/pneumothorax-segmentation

Models trained at 768x768 (or close to that) using AdamW optimizer. For the FPN models Flip TTA was used whilst Scale (1024) TTA was used for the rest. We used two thresholds, one for segmentation and another (of higher value) for classification

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107872

Solution

8th solution

Summary Paper :

https://docs.google.com/document/d/108xK_J4WVTtuuwMfxBINR7FLXYRNxldnAQV5cLf0-1g/edit

Code :

https://github.com/i-pan/kaggle-siim-ptx

Solution

8th solution

- Data split: 10% holdout for ensemble, 10-fold CV on remaining 90%, stratified by pneumothorax size
- Architecture: DeepLabV3+
- Backbone: ResNet50/101 and ResNeXt50/101 with group normalization
- Loss: weighted BCE (trained on all images) or soft Dice (trained on positives only)
- Optimizer: Vanilla SGD, momentum 0.9

- Training: batch size 4, 1024 x 1024 and batch size 1, 1280 x 1280 for pure segmentation (did not retrain on stage 2)
- Schedule: cosine annealing, 100 epochs, 5 snapshots, initial LR 0.01 to 0.0001

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107522

Solution

8th solution

Ensemble:

- 12 models total (x3 snapshots/model)
- 4 trained on positives only with soft Dice
- 8 trained on all images with weighted BCE
- I used 4 of the models trained on all images as "classifiers"
- Max pixel value was taken as classification score, averaged across 4 models
- Multiplied pixel-level scores from 4 models trained on positives only by this classification score, then averaged
- Final ensemble: multiplied score as above averaged with pixel-level scores based on other 4/8 models trained on all images
- Hflip TTA

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107522

2. Solution

(8) 8^{th} Solution

Solution

8^{th} solution

Ensemble:

- **Max pixel** value was taken as classification score, averaged across 4 models

Ian Pan Topic Author • (8th in this Competition) • 2 years ago • Options • Report • Reply

4

We used 0.7 as our threshold (this was after it was multiplied by the segmentation output from the model trained on positive images only, so it is a bit lower). For our single stage models, the best classification threshold was around 0.8 with segmentation threshold at 0.4.

I think the trick was proper weighting of the BCE loss function. We actually used your loss function, modified slightly (thanks!):

```
loss = (pos_frac*pos*loss/pos_weight +
neg_frac*neg*loss/neg_weight).sum()
```

In your original loss function, `pos_frac = 0.25` and `neg_frac = 0.75`. I modified it so that `pos_frac = 0.1` and `neg_frac = 4.9`. I also used `neg_frac = 7.9` for some models.

출처 : https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107522

Solution

8th solution

Squeeze-and-Attention Networks for Semantic Segmentation
https://arxiv.org/pdf/1909.03402.pdf

Ensemble:

- **Max pixel** value was taken as classification score, averaged across 4 models

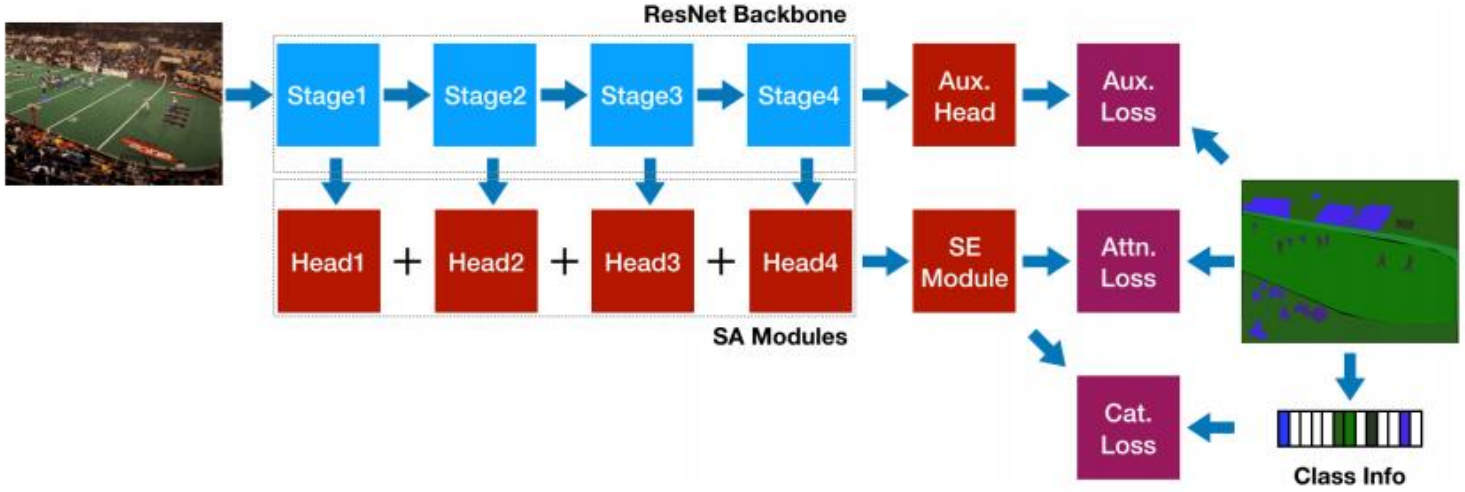


Figure 3: Squeeze-and-attention Network. The SANet aggregates outputs from multiple hierarchical SA heads to integrate multi-scale features instead of extracting them from the last stage. To regularize the training process, in addition to the attention loss, we employ two losses to take image-level categorization and pixel-level dense prediction into consideration. The auxiliary head designed for auxiliary loss is composed of fully convolutional layers. The SE head designed for categorical loss has a structure of a SE module. In this way, we utilize the pixel-group attention extraction capacity of SA modules and integrate multi-scale contextual features simultaneously.

Solution

9th solution

Code : https://github.com/scizzzo/kaggle-siim-pneumothorax

Classification :

- Backbone: seresnext50
- Data: For this step I used all images and balanced batches(pneumothorax/non-pneumothorax) it greatly accelerated convergence
- Data splits: 5 folds and 10 folds stratified by pneumothorax area.
- Input size: 768x768
- Loss: BCE
- Augmentations: hflips, rotations(up to 10 degree), random brightness, contrast and gamma, blur
- Lr scheduling: reduce lr on plateau with patience=3 epochs.

Solution

9th solution

Segmentation :

- Backbone: seresnext50
- Data: only images containing pneumothoraxes
- Data splits: 5 folds and 10 folds stratified by pneumothorax area
- Input size: 928x928, 768x768
- Loss: BCE + Dice
- Augmentations: same as in the classification stage
- Lr scheduling: reduce lr on plateau with patience=5 epochs.

Solution

10th solution

Code : https://github.com/SgnJp/siim_acr_pneumothorax

Common :

- Data split: CV5
- Optimizer: Adam
- Scheduler: Reduce lr on plateau
- Augmentations: relatively aggressive: ShiftScaleRotate, Grid- and Elastic-transformation, GaussianNoise

Solution

10th solution

Classification :

- Models: se_resnext101 (2 snapshots), senet154
- Resolution: 768x768
- Loss: BCE
- Additional features: TTAx2(hlip), pseudo labeling, gradient accumulation (bs = 100-200)

- The key here is the model selection. Instead on focusing on accuracy, I focused on f0.5 metric with fixed threshold of 1.0. The motivation is quite simple, the average dice of segmentation model is around 0.58. It means that correctly guessed negative instance contributes to the score with 1, and correctly guessed positive instance only with 0.58.

Solution

10th solution

Segmentation :

https://arxiv.org/abs/1704.00109

- Models: Unets: dpn98, se_resnet101, se_densenet121 (2 snapshots of each)
- The train process consist of 4 stages:
  - Loss: BCE + dice; size: 512x512
  - Loss: BCE + dice; size: 1024x1024
  - Loss: BCE; with soft pseudo labels; size: 1024x1024
  - Loss: symmetric lovasz; size: 1024x1024
- Additional features: TTAx6(hlip + rescale), gradient accumulation (bs = 50-100)