Automatic Detection of Brain Aneurysms: segmenting CT scans using CNNs

Jason Qin, Harry Emeric
{Department of Chemical Engineering, Department of Statistics}, Stanford University
jzqin@stanford.edu, harryem@stanford.edu

Abstract

Deep Learning currently presents many exciting applications to medical imaging research. This project seeks to explore how CNNs can automatically detect and segment ruptured intracranial aneurysms from CT angiography (CTA) scan images. We train a semantic segmentation model using a ResNet50 encoder [1] with a selection of decoders [2][3][4]. We investigate how these various architecture combinations perform under a number of hyperparameters, and evaluate these results in terms of their practical use in a clinical setting.

1 Introduction

Brain aneurysms affect 1-3% of adults, and aneurysm rupture has a high mortality rate, making their detection once ruptured very time sensitive. Furthermore, detection of aneurysms from CTA scans is a time intensive task. A method of rapid detection in emergency settings would be highly valuable, and deep learning approaches offer some ways to achieve this; convolutional neural networks (CNNs) have shown great promise in applying computer vision techniques to medical problems. Using a set of CTA scan images along with corresponding segmentation masks labeling the aneurysms, we train a semantic segmentation network based on ResNet50 followed by either a convolutional, Pyramid Pooling (PPM) with convolutional upsampling, or Unified Perceptive Parsing Network (UPerNet) decoder. The trained network is used to predict a segmentation mask given a new CTA scan. An emergency condition can therefore instantly be identified and prioritized. Typically, these architectures have been used for multilevel pixel-wise classification (semantic segmentation) [2][3][4], in which each image has many classes of objects to segment simultaneously [1]. We seek to understand how these networks capture features of CT scans for the segmentation of aneurysms.

2 Related Work

Feature segmentation has been applied to numerous biomedical problems [5]. For instance, CNNs have been used for brain tumor segmentation [6], pulmonary embolism detection [7], and knee cartilage segmentation [8]. The field is rapidly evolving, with advances in computer vision improving medical image diagnostics. Recent unpublished work in the Yeom lab has incorporated UNets [9] to tackle a similar problem: detection of unruptured aneurysms.

3 Dataset

Our dataset consists of 57 patients' CTA scans, collected by Professor Kristen Yeom in the Stanford Medical School. Each patient has around 500 CT images each, with slices spanning from the top of the head to the clavicle (or vice versa). For a given patient, around 5-10 slices contain aneurysms. For each slice, the Yeom lab produced segmentation masks, which provide the corresponding set of pixels in each image which contain an aneurysm. The CT images are provided as 512 x 512 images stored in the in Digital Imaging and Communications in Medicine (DICOM) format. Segmentation masks were produced using the software program ITK-SNAP [10], and stored as Nearly Raw Raster Data (NRRD) files.

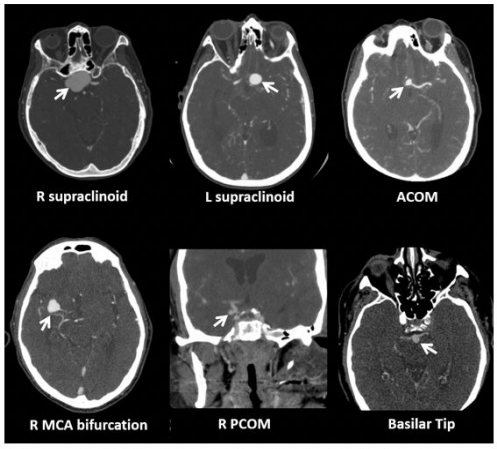

Figure 1: Examples of aneurysms detected by CTA scans; aneurysms are labelled with white arrows

Label	Pixels	Percentage
Background	10^9	99.9976%
Aneurysm	~160K	0.00235%

Table 1: Class imbalance within dataset

Preprocessing involved converting these DICOM and NRRD files into PNGs, and matching segmentation mask images to their corresponding CT scan slices. Each DICOM image is stored as a RBG PNG with dimensions of 512 x 512 x 3, and each NRRD mask is stored as a 512 x 512 grayscale image. The data were split with as 51 and 6 patients in the train and test sets, respectively.

3.1 Data Augmentation

Data were augmented through random horizontal flips, rotations, and cropping to image dimensions of [300, 375, 450].

3.2 Class Imbalance Resampling

The patient data contains significant class imbalance. Only 1- 2% of scan slices contain an aneurysm, and across all pixels in the dataset, over 99.998% are background, and 0.002% are aneurysm (**Table 2**). The sparsity of the aneurysm data is augmented by increasing the positive aneurysm data set 100-fold by randomly resampling scans with aneurysms from across all patients, with the resampled scans shuffled randomly into the dataset. These resampled images also underwent data augmentation.

4 Methods

Our semantic segmentation model incorporates two architectures: an encoder to map images onto a lower-dimensional feature space, and a decoder to upsample features to pixel-wise classifications. We use ResNet50 [1] as the encoder, and vary the decoder between a convolutional bilinear upsampling architecture, a Pyramid Pooling Module (PPM) following by convolutional upsampling described in [3], and a Unified Perceptual Parsing (UPerNet) network [4]. We keep the encoder constant, using pretrained weights from the MIT ADE20K dataset [2]. We vary the decoder because decoders are often not pre-trained, and as such may contribute greater variability to model results.

4.1 Metrics

4.1.1 Loss Function

The objective of this segmentation algorithm is to correctly classify as many pixels as possible as "containing aneurysm". We use the weighted Negative Log Likelihood Loss (NLLL) [11], with n batches over c output classes:

$$\ell(x, y) = L = \{l_1, \dots, l_N\}^\top, \quad l_n = -w_{y_n} x_{n,y_n} \qquad w_{y_c} = \frac{1}{pixels\ per\ class\ c}$$

where weights can modify the strength of loss for any class. Because of the class imbalance, each class weight was originally scaled relative to the inverse of the number of pixels present for each class, so the loss function is more heavily weighted by aneurysm loss.

4.1.2 Evaluation Metric

The evaluation metric used is aneurysm $IOU = \frac{|X \cap Y|}{|X \cup Y|}$, where X are ground-truth aneurysm pixels, and Y are predicted aneurysm pixels. Aneurysm IOU measures the overlap between prediction and ground truth for aneurysm pixels, divided by the union of prediction and ground truth.

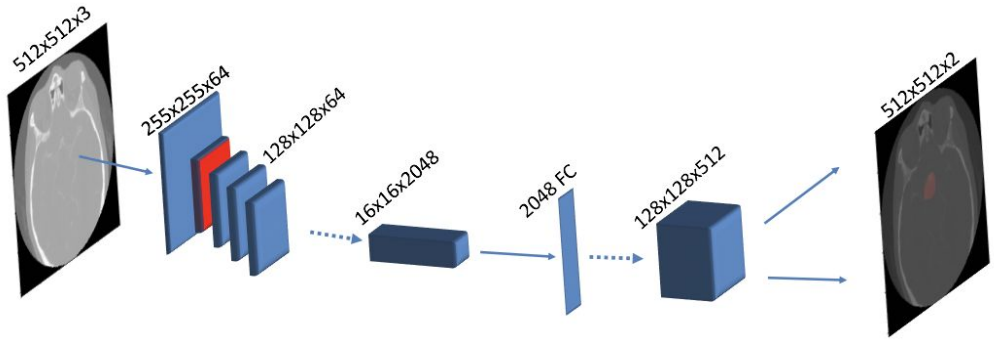

Figure 2: Schematic of ResNet50 decoder with convolutional upsampling. The image first passes through a decoder, reducing its dimensionality. Next, the reduced-dimensional representation of the image is passed through the decoder, generating pixel-wise aneurysm predictions. The full architecture descriptions can be found in [1] and [2].

4.2 Architecture

4.2.1 Encoder

ResNet50: ResNet50 from [1] is adapted, with the final fully connected layer dimension set to 2048. Briefly, the ResNet50 encoding architecture runs the inputs through a 7x7, 64 convolutional layer and a 3x3 max pool layer, and then into a series of 4 types of "bottleneck" block which are applied 3,4,6 and 3 times, before being fed into a fully connected layer (see **Figure 2**).

4.2.2 Decoder

Convolutional: Convolutional upsampling uses two 3x3 upconvolutional, batch-normalized, ReLU layers to convert from the 2048-dimensional fully connected layer output by ResNet50 to a 512x512x2 (since 2 classes) matrix, with pixel labels predicted using softmax [2] (see **Figure 2**).

PPM: Pyramid pooling [3] combines features under four pyramid dimensions, which represent the image using receptive fields of different sizes. The pyramid layers are upsampled to the encoder feature map size, and concatenated before being put through a upsampling convolutional layer, which outputs the final prediction.

UPerNet: Unified Perceptive Parsing Network (UPerNet) [4] uses a Feature Pyramid Network (FPN) with a PPM applied to the encoder output before feeding into the FPN. The decoder FPN incorporates lateral connections from the encoder during upconvolution to improve decoder pixel annotation accuracy.

5 Experiments/Results/Discussion

To determine the best model to predict aneurysms, we iteratively improve our results by 1) investigating different architectures, and 2) tuning hyperparameters for the best model. We use the following settings for all investigations: stochastic gradient descent (SGD) optimizer; momentum $\beta = 0.9$; batch size = 4; L2 weight decay factor = 0.0001; and train for 20 epochs on data. The encoder was loaded with pretrained weights from MIT ADE20K.

5.1 Architecture Search

For the architecture search, we use the same parameters specified above for each architecture, and in addition set the learning rate α = 2e-2. None of the architectures predicted any aneurysms without significant positive aneurysm image resampling, so we performed an initial 40-fold resampling, increasing the number of positive aneurysm images from ~600 to ~24,000. Of the three architectures tested (ResNet50 + Conv, ResNet50 + PPM, ResNet50 + UPerNet), ResNet50 + UPerNet maximized IOU in both the training and test sets (**Table 2**). UPerNet outperformed the other decoders, suggesting that using a FCN framework that detects features at different semantic levels (for instance, Head → Brain → Blood Vessel → Aneurysm) is superior to simpler models.

Despite the inclusion of regularization techniques like L2 weight decay and batch normalization, all three architectures had large variances, with IOU 2- to 10-fold greater for training IOU than eval IOU. This suggests that

Encoder	Decoder	Training IOU	Test IOU	Time per Epoch (s)
ResNet50	Conv	1.67e-2	1.8e-3	**~270**
	PPM	9.0e-3	1.3e-3	~360
	UPerNet	**3.15e-2**	**1.41e-2**	~540

Table 2: Analysis of ResNet50 + Conv/PPM/UPerNet architectures. The UPerNet decoder outperforms both the convolutional upsampling and PPM architectures on both training and evaluation IOU. However, it takes roughly 2x as long to run.

Hyperparameter	Value	Training IOU	Test IOU
Upsampling	10-fold	0.0272	0.0101
	40-fold	0.0500	0.0237
	100-fold	**0.0662**	**0.0252**
Background Loss Weight	5.2e-4	0.050	0.0025
	5.2e-3	0.209	0.237
	5.2e-2	**0.310**	**0.287**
Learning Rate	2e-3	0.316	0.384
	2e-2	0.352	0.314
	2e-1	**0.357**	**0.419**

Table 3: Analysis of hyperparameter tuning. Upsampling analysis was done using background weight loss of 5.2e-4 and learning rate of 2e-2. Background loss weight analysis was done with 40-fold upsampling and learning rate of 2e-2. Learning rate analysis was done with 100-fold upsampling and background weight loss of 5e-2. All hyperparameters were tested for 20 epochs, and the training IOU is the average IOU of the training data over the last epoch. The test IOU was calculated using the model generated after the 20 epochs of training.

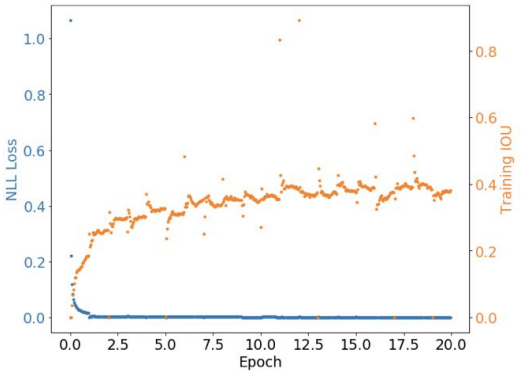

Figure 2: Schematic of ResNet50 decoder with convolutional upsampling. The image first passes through a decoder, reducing its dimensionality. Next, the reduced-dimensional representation of the image is passed through the decoder, generating pixel-wise aneurysm predictions. The full architecture descriptions can be found in [1] and [2].

the data were overtrained with the parameters used, which is likely because all 24,000 positive image training examples originate from a pool of 600 positive aneurysm images.

5.2 Hyperparameter Tuning

After determining the best architecture, we examined the impact of several hyperparameters on model results. We vary the hyperparameters independently, and combine the best parameters in each analysis for the final model. We examined the following hyperparameters: extent of upsampling, model loss weights, and learning rates. For loss weights, we started with $w_{y_c} = \frac{1}{\#\ pixels\ per\ class}$, which provided loss weights of 5.2e-4 and 0.9995 for the background and aneurysm pixels, respectively. Tested parameters and results are shown in **Table 3**, and show that the best parameters were upsampling 100-fold, using w_{y_c} = [5.2e-2, 0.95], and α = 2e-1. Upsampling by 100-fold increased training and evaluation IOU by around 3-fold; increasing the loss weight of background pixel increased training IOU by ~6-fold, and increased evaluation IOU by over 10-fold; learning rate had a minor impact on IOU.

With the best hyperparmeters, the model no longer has a variance problem, as the test IOU (0.419) is greater than the training IOU (0.357). These IOU values are $O(10^4)$ better than random-chance IOU, which is calculated as $\frac{\#\ aneurysm\ pixels * P(aneurysm)}{2*(\#\ aneurysm\ pixels)} \approx 10^{-5}$. Notably, the variance between test IOU and training IOU decreases after increasing the loss weight of the background pixels. This suggests that the solution space is more accurately modeled when the background pixels are given more weight, i.e. the aneurysm pixels are not overweighted and thus not overfitted. The training loss and IOU of the best model are shown in **Fig 2**.

5.3 Model Applicability

The applicability of this model in a clinical setting is the most relevant success metric. Failing to detect an aneurysm at all is a serious problem, but false positive predictions waste radiologist time. We decide to compare the final model to a model with better recall, which was trained with 40-fold upsampling, a background loss weight of 5.2e-4, and α = 2e-2. The improved-recall model had a training IOU of 0.050, and a test IOU of 0.0025.

As shown in **Table 4**, the final model suffers from poor recall: it falsely labels some aneurysms as background. Empirically, the final model correctly predicts aneurysm location in a subset of every patient's slices, meaning the radiologist would see a positively labeled slice for each patient. However, the low recall makes the final model potentially dangerous to use in practice. Meanwhile, the improved-recall model had perfect recall, but suffers from poor precision, likely generating too many false predictions to be useful practically.

Final Model (background weight loss = 5.2e-2)				Improved-recall model (background weight loss = 5.2e-4)			
	Aneurysm Predicted	Aneurysm Not Predicted	Recall		Aneurysm Predicted	Aneurysm Not Predicted	Recall
Aneurysm Present	39	21	0.65	Aneurysm Present	**60**	0	**1**
Aneurysm Not Present	23	2817		Aneurysm Not Present	943	1897	
Precision	**0.63**			Precision	0.06		

Table 4: Image-level analysis of aneurysm detection for the 6 patients in the test set. There are 60 total CT scan slices with aneurysms in the test set, and 2900 images scan slices total. The aneurysm predicted/aneurysm present box shows the number of images in which the location of the aneurysm is included in the predicted pixels for the model's output. The precision and recall of aneurysm prediction are calculated for both the final model (background loss weight = 5.2e-2) and a model that better performs on recall (background loss weight = 5.2e-4). The recall-favored model suffers from worse precision, meaning that radiologists would waste time parsing model outputs. However, the final model suffers from poor recall: some aneurysms are missed entirely, and thus the model may be misleading for a radiologist using it as an aid.

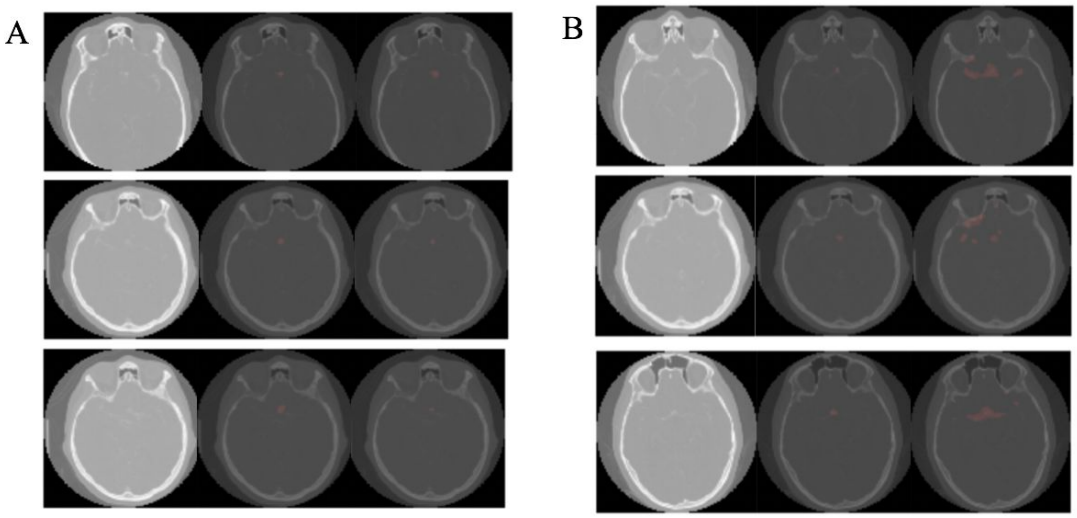

Figure 4: Examples of aneurysm localization using the A) final model and B) improved-recall model. The final aneurysm model is significantly better at localizing aneurysms than the improved-recall model. The improved-recall model locates each aneurysm, but also predicts as aneurysm areas several times larger than the aneurysm itself.

An analysis of the images (**Figure 4**) show that the improved-recall model manages to locate aneurysms, but performs far worse than the final model at precise localization. The low IOU of the improved-recall model occurs both because of false positives in images without aneurysms, as well as overprediction in pictures that contain aneurysms.

6 Conclusion and Future Work

ResNet50 + UPerNet may assist radiologists in localizing aneurysms within CT angiogram scans of the human brain. The ResNet50 + UPerNet model, by using a FCN with a PPM head, outperforms simple upconvolutional and PPM with upconvolution decoder models. The best architecture and hyperparameters chosen generate a test IOU of 0.419 on a data set that has a severe [99.998%, 0.002%] class imbalance, representing a 10,000-fold improvement over random chance IOU. Although the final model has imperfect recall, it may be a useful first-pass tool for radiologists to locate obvious aneurysms.

In the future, more network architectures should be tried for this challenging segmentation task. Examples include UNet [12], and recently developed in-house models within the Yeom lab, such as XNet. A 3D model, where slices are fed in batches, could detect dependencies between neighbouring images from the same patient. Future work should also include collecting more data, in order to better model the space of possible aneurysms, which will likely improve prediction ability.

Contributions

Both Harry and Jason coded the preprocessing pipeline and worked on model evaluation & analysis. Jason was responsible for data augmentation, architecture and hyperparameter search, and training the models. Both parties worked collaboratively on finding a worthwhile problem, presenting the poster, and writing the reports.

References

[1] He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian (2015-12-10). "Deep Residual Learning for Image Recognition". arXiv:1512.0338

[2] Semantic Understanding of Scenes through ADE20K Dataset. B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso and A. Torralba. arXiv:1608.05442. (https://arxiv.org/pdf/1608.05442.pdf); https://github.com/CSAILVision/semantic-segmentation-pytorch: Scene Parsing through ADE20K Dataset. B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso and A. Torralba. Computer Vision and Pattern Recognition (CVPR), 2017.

[3] Zhao, Hengshuang, et al. "Pyramid scene parsing network." *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[4] Unified Perceptual Parsing for Scene Understanding. T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. European Conference on Computer Vision (ECCV), 2018. (https://arxiv.org/abs/1807.10221)

[5] Tajbakhsh, Nima, et al. "Convolutional neural networks for medical image analysis: Full training or fine tuning?." *IEEE transactions on medical imaging* 35.5 (2016): 1299-1312.

[6] M. Havaei, *Brain tumor segmentation with deep neural networks*, 2015, [online] Available: arXiv:1505.03540.

[7] N. Tajbakhsh, J. Liang, "Computer-aided pulmonary embolism detection using a novel vessel-aligned multi-planar image representation and convolutional neural networks", *Proc. MICCAI*, 2015.

[8] A. Prasoon, K. Petersen, C. Igel, F. Lauze, E. Dam, M. Nielsen, "Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network", *Proc. MICCAI*, pp. 246-253, 2013.

[9] Kayalibay, Baris, Grady Jensen, and Patrick van der Smagt. "CNN-based segmentation of medical imaging data." *arXiv preprint arXiv:1701.03056* (2017) https://arxiv.org/abs/1701.03056

[10] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage* 2006 Jul 1;31(3):1116-28.

[11] https://pytorch.org/docs/stable/_modules/torch/nn/modules/loss.html#NLLLoss

[12] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015