Pulmonary Embolism Classification in Lung CTA

Jananan Mithrakumar
Department of Electrical Engineering
Stanford University
janamith@stanford.edu
Video Link: https://www.youtube.com/watch?v=_duRCnwAkE
Github Link: https://github.com/Janamith/CS230-Project

Abstract

Pulmonary Embolism (PE) is a blockage in one of the pulmonary arteries which is regularly diagnosed using computed tomography angiography (CTA). Untreated, PE is associated with a significant mortality rate (as high as 30%) whereas the mortality rate of diagnosed and treated PE is around 8%. Therefore, accurate diagnosis of PE using CTA is very important for patient outcomes. Traditionally, PE in CTA scans have been detected through inspection by trained radiologists.However, existing challenges include inter-grader variability and high false-positive and false-negative rates. We propose a deep learning approach using ConvNets to detect the existence of PE in CTA slices within CTA scan volumes.

1 Introduction

Due to the importance of early diagnosis of PE in reducing the mortality of patients, detection of PE using CTA is very important for patient outcomes. However, traditional CTA scan examination by radiologists leads to problems such as inter-grader variability, high false-negative and high false-positive rates. Use of state-of-the-art ConvNet based deep learning approaches for detection of lung cancer and other pulmonary diseases with visual morphology have been quite successful. PE shares many characteristics with these diseases which implies the possibility of a similar approach being successful.

We propose a deep learning approach using ConvNets to try and alleviate these issues. This led As a first step into investigating the feasibility of using deep learning to detect PE in CTA scans we use a 2D ConvNet approach. As such, the inputs to our model are 2D slices extracted from CTA volumes. The output of our model is a binary classification; whether the slice contains PE or does not contain PE.

2 Related work

Previous work into the automated detection of PE in lung CTA primarily consisted of the use of various supervised learning techniques to learn on hand-extracted features. These features were extracted from the CTA scans using various image processing techniques.

Two examples of this technique are the paper by Bouma H et al[1] and the paper by Ozkan et al[2]. Bouma et al followed a procedure where segmentation and candidate detection techniques were used to reduce the search area to a smaller collection of CTA regions which have a high chance of containing an embolus. Once this was done, features such as intensity, shape, location and size of candidate regions were extracted. These features were then used with various classifiers such as a bayes norm classifier and a decision tree and the performances were compared. Using this approach,

,

an average sensitivity of 70% was obtained which is not too great. In the paper by Ozkan et al, candidate PE regions had already been delineated by radiologists from CTA scans. Feature extraction was then performed on the candidate regions and fed into various classifiers including an ANN with 2 hidden layers. With this approach, they were able to achieve a sensitivity of 97%. However this approach still relies on the detection of candidate regions by a radiologist and hand-extracted features.

Deep learning approaches have primarily been taken in the automated detection of lung cancer in CTA scans. The paper by Ardila et al[3] is one example of such an approach. In this paper, various state-of-the-art DL techniques were combined to form a cancer risk detection system. Firstly, a mask-RCNN was used to perform lung-segmentation. Secondly, a cancer ROI detection model using a RetinaNet architecture was used to predict ROIs. Thirdly, a full CTA volume model using 3D inflated Inception V1 was used to extract full volume features. Finally, another inception network was trained on features extracted from the previous models to output a final prediction. The techniques used in this paper are state-of-the-art and served as inspiration for the techniques that we explore here.

3 Dataset and Features

The dataset we used is the FUMPE (Ferdowsi University of Mashhad's PE) dataset[4]. This dataset is a public dataset of CTA scans of 35 patients amounting to a total of 8792 slices. Each slice image is a 512px by 512px 2D image. The ground-truth images are also provided, where the PE regions have been precisely delineated in every slice of each CTA by an expert.

Due to the small number of slices that the dataset provides, we perform data augmentation to boost the number of examples. Namely, each slice is flipped, mirrored and rotated to quadruple the number of examples. Before training, the 2D slice images are also downsampled to 256px by 256px.

With the above data augmentation we are left with 35168 examples. This dataset is then split 70/20/10 to produce the training/validation/test sets.

Labels were created by looking at which slices in the ground truth images contained annotated PEs.

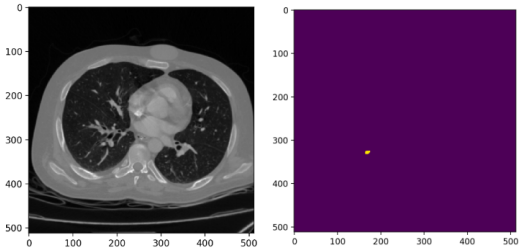

Figure 1: Dataset example Left: Base CTA slice image Right: Ground truth mask – yellow region is a PE

4 Methods

4.1 Baseline model

As a baseline model we used a LeNet architecture with binary cross-entropy loss.

$$L(y,\hat{y}) = -\frac{1}{N}\sum_{i=0}^{N}(y * log(\hat{y}_i) + (1-y) * log(1-\hat{y}_i))$$

Figure 2: Binary cross-entropy loss

4.2 ResNet50

To improve on the previous model we transitioned onto a deeper more sophisticated model, a ResNet with 50 layers. Binary cross-entropy loss was used.

4.3 Focal Loss

To deal with the sparsity of positive examples in the training set, we decided to use focal loss instead of binary cross-entropy loss. Focal loss penalises the contribution of the easily classified negative examples and increases the contribution of the positive examples to the loss.

$$FL(p_t) = -(1-p_t)^{\gamma} log(p_t) \quad (1)$$

$$p_t = \begin{cases} p, & \text{if y=1} \\ 1-p, & \text{otherwise} \end{cases} \quad (2)$$

Figure 3: Focal Loss

4.4 InceptionV1

After experimentation with LeNet and ResNet50, we observed that the representational power of neither networks was great enough to produce the sensitivity we needed. Therefore, we moved onto the InceptionV1 architecture.

4.5 Lung segmentation -> Classification

Finally, to further improve accuracy and sensitivity we performed lung segmentation on the CTA slice images.

Lung segmentation is performed using a pre-trained U-net model[5] that produces a left and right lung mask for each CTA slice. The masks are then applied to each CTA slice to extract the left and right lung regions. The regions are then compared to the ground truth images to find in which lung/s the PE is situated. Two images are then produced with one of the lung regions and the appropriate label.

The augmented dataset was then used to train an InceptionV1 model.

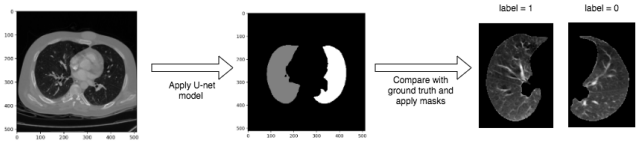

Figure 4: Lung segmentation process

5 Experiments/Results/Discussion

5.1 Baseline Model

For our LeNet model we chose to use an Adam optimizer with a mini-batch size of 32. We trained the model for 5 epochs. We decided on this mini-batch size by performing a hyper-parameter search over mini-batch sizes of 4/16/32/128. Training and validation accuracy stabilized after 5 epochs which was why the model was trained for that many epochs.

Our LeNet model achieved the following results:

train set accuracy = 85.23%
test set accuracy = 74.63%

	Predicted Positive	Predicted Negative
Actual Positive	22	210
Actual Negative	13	634

Table 1: Test set Confusion Matrix

The model achieves good train set accuracy and decent test set accuracy. However from the confusion matrix we can see that this is mainly due to the sparsity of positive examples which the model

predicts incorrectly almost every single time (false negative rate of 90.52%). It is likely that the LeNet architecture did not have enough parameters and layers to learn the features required for this classification task resulting in high bias. We therefore decided to move onto a deeper neural network architecture.

5.2 ResNet50

We trained a ResNet50 model with an Adam optimizer and a mini-batch size of 32 (found through parameter search). The train and validation set accuracy stabilized after training for 5 epochs.

Our ResNet50 architecture produced the following results:

train set accuracy = 82.34%
test set accuracy = 74.52%

	Predicted Positive	Predicted Negative
Actual Positive	48	184
Actual Negative	40	607

Table 2: Test set Confusion Matrix

This deeper model also has good train set and test set accuracy. When we look at the confusion matrix we see that the false negative rate has reduced quite substantially to around 80%. However, this false negative rate is still quite high.

5.3 Focal Loss

To try and reduce the false negative rate further we changed our loss function to focal loss instead of binary cross entropy loss. We used the same hyper-parameters as before. With focal loss we obtained the following results:

train set accuracy = 80.43%
test set accuracy = 77.13%

	Predicted Positive	Predicted Negative
Actual Positive	83	149
Actual Negative	52	595

Table 3: Test set Confusion Matrix

With focal loss, we get a even better false negative rate of around 64%. However it became apparent that a even deeper network was required to classify the hard positive examples as the models still suffered from high bias.

5.4 InceptionV1

We decided to train an Inception V1 model with an Adam optimizer and mini-batch size of 32. We trained the model for 20 epochs.

Our Inception V1 model produced the following results:

train set accuracy = 90.51%
test set accuracy = 82.48%

	Predicted Positive	Predicted Negative
Actual Positive	110	122
Actual Negative	32	615

Table 4: Test set Confusion Matrix

With this far deeper model we got better accuracy and sensitivity. The false negative rate dropped further to around 48%. However, for medical imaging purposes, the sensitivity of the model was still far to low to be considered effective.

5.5 Lung segmentation –> Classification

To try and further improve the accuracy and sensitivity of our model, we perform lung segmentation on the input images before feeding them into our model.

With lung segmentation and our Inception model we get the following results:

train set accuracy = 89.77%
test set accuracy = 82.17%

	Predicted Positive	Predicted Negative
Actual Positive	112	120
Actual Negative	35	612

Table 5: Test set Confusion Matrix

Performing lung segmentation on the input images did not lead to any improvement to accuracy and sensitivity as we had expected.

This could be due to the non-variability of the parts of the CTA images surrounding the lung regions which led our models to learn filters that effectively ignore these areas in the images.

6 Conclusion/Future Work

We obtained the best results using the InceptionV1 model. This was expected since the InceptionV1 model had a far greater number of layers and parameters than the other models. Lung segmentation on the input images did not improve the accuracy or sensitivity of the model.

Regardless, none of the models were able to produce high enough sensitivities to be considered effective for the classification task. We hypothesise that this is due to the relatively large variability of PE shape, size and location across the different CTA images coupled with the small number of hard positive examples used during the training of the models. Ideally, we would have approached this problem via a 3D ConvNet model that took CTA volumes instead of CTA slices into consideration. However, the dataset used was not nearly large enough to train such a model.

Future work would first involve collecting a much larger dataset and re-training the models used above, perhaps trying more state of the art ConvNets such as InceptionV3 as well. Once this is done classification based on CTA volumes versus single slices could be explored. This could be performed using a 3D inflated Inception model for example. Another approach that could be explored would be a ROI approach where a model is developed that could produce a bounding box localising the PE.

References

[1] Bouma, H., Sonnemans, J., Vilanova, A., Gerritsen, F. (2009). Automatic Detection of Pulmonary Embolism in CTA Images. IEEE Transactions on Medical Imaging, 28(8), 1223-1230. https://doi.org/10.1109/tmi.2009.2013618

[2] Özkan, H., Osman, O., Şahin, S., Boz, A. F. (2014). A novel method for pulmonary embolism detection in CTA images. Computer Methods and Programs in Biomedicine, 113(3), 757-766. https://doi.org/10.1016/j.cmpb.2013.12.014

[3] Ardila, D., Kiraly, A. P., Bharadwaj, S., Choi, B., Reicher, J. J., Peng, L., Tse, D., Etemadi, M., Ye, W., Corrado, G., Naidich, D. P., Shetty, S. (2019). Author Correction: End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography. Nature Medicine, 25(8), 1319-1319. https://doi.org/10.1038/s41591-019-0536-x

[4] asoudi, M., Pourreza, H., Saadatmand-Tarzjan, M., Eftekhari, N., Zargar, F. S., Rad, M. P. (2018). A new dataset of computed-tomography angiography images for computer-aided detection of pulmonary embolism. Scientific Data, 5(1). https://doi.org/10.1038/sdata.2018.180

[5] Automatic lung segmentation in routine imaging is a data diversity problem, not a methodology problem. (n.d.). arXiv.org. https://arxiv.org/abs/2001.11767