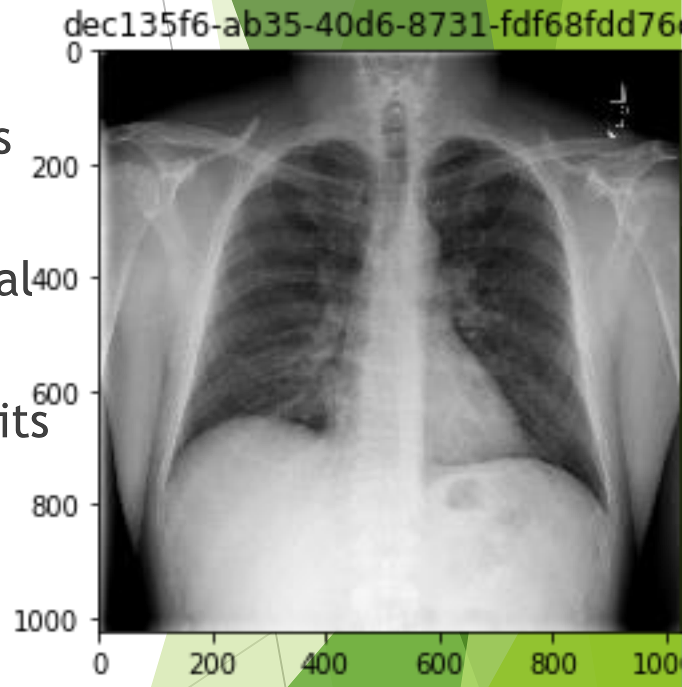
Classifying chest radiographs with CNNs

DATS 6501 Capstone

Vinay Bhandaru

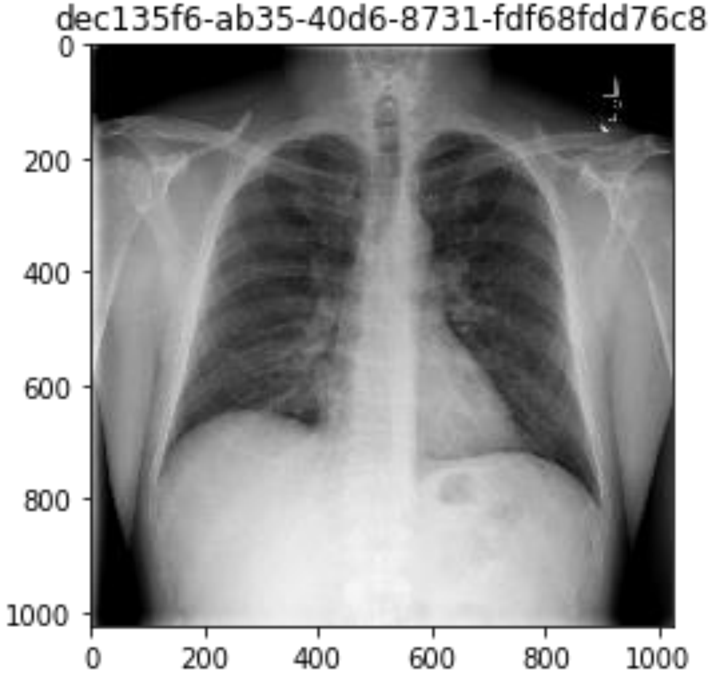
Background

- In an effort to improve diagnostic services the Radiological Society of North America (RSNA) created a database of over 25,000 labeled chest radiographs and provided it to Kaggle’s machine learning community.
- They recognize the potential for ML to automate initial detection of potential pneumonia cases in order to prioritize and expedite their review.
- In the United States in 2015 pneumonia accounted for over half a million visits to the ER and over 50,000 deaths.
- The objective of this project is to determine how well chest radiograms can be classified as normal or abnormal or containing lung opacities using convolutional neural networks.

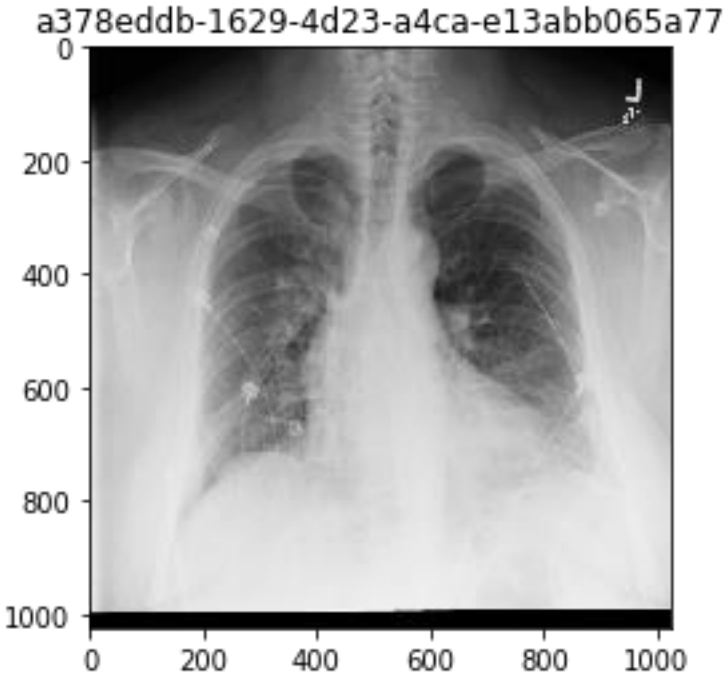


3 classes of chest x-rays

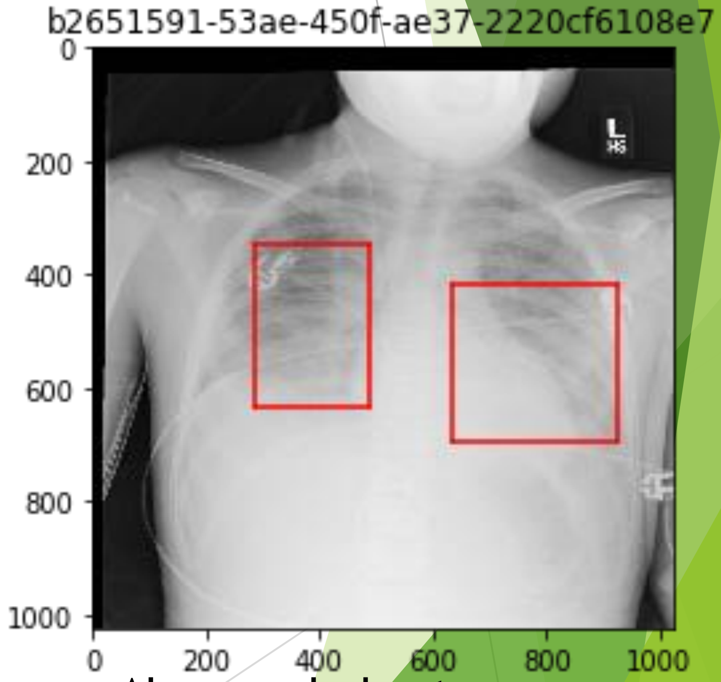
-The data consists of 25,684 1024X1024 grayscale chest radiograph images in dicom format (.dcm)
-8525 images are normal and 17,159 are abnormal.
-Among the abnormal images, 5659 have regions of opacity



Normal chest x-ray

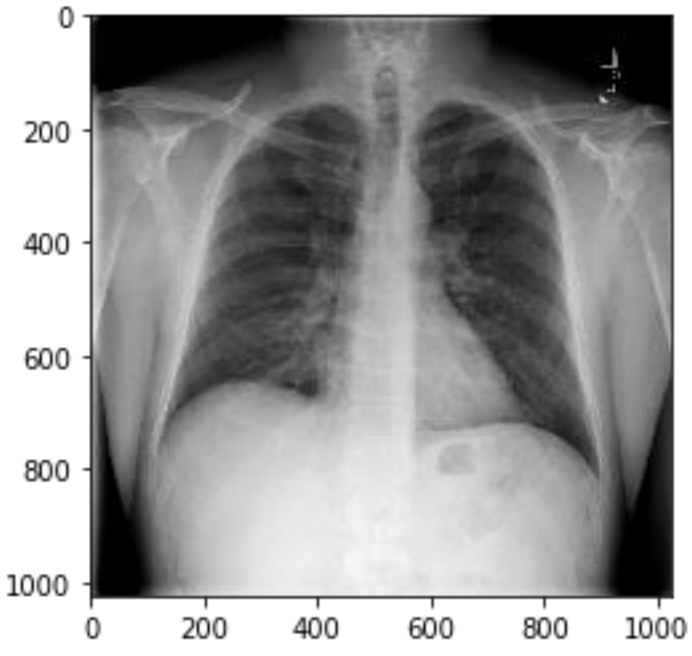


Abnormal chest x-ray without regions of opacity



Abnormal chest x-ray with regions of opacity indicated by bounding boxes

Preprocessing images



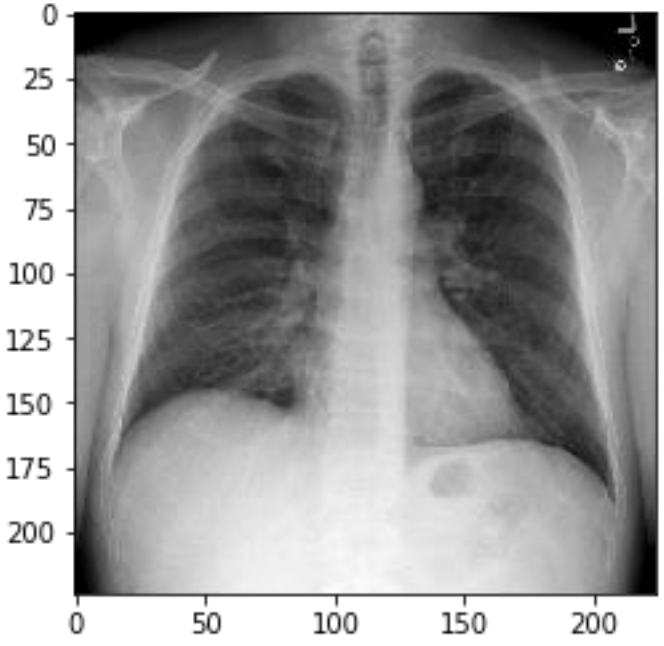


Image on left is original 1024X1024 image and image on the right is result of cropping1024X1024 image to 896X896 and then sampled every 4th pixel to resize to 224X224. In addition, images with one color channel were converted to 3 color channels.

The images were subsequently normalized based on the mean and standard deviation of images in the ImageNet training set which are mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225]

CNNs

- Convolutional neural networks have become popular for image recognition in recent years since they make better use of spatial information in an image through the use of convolution filters.
- Many different CNN architectures have been implemented, including variations of ResNet, AlexNet, and VGG.
- CNNs are often pre-trained on large image databases, such as ImageNet, instead of being randomly initialized and trained de novo. Then they are fine-tuned on the dataset of interest. In general this process reduces training time, promotes model convergence, and reduces overfitting.
- This is the approach that is followed by Rajpurkar in the development of the CheXNet algorithm which is an algorithm designed to detect pneumonia from chest x-rays using DenseNet-121. The final fully connected layer of DenseNet-121 was replaced with one that has a single output. The weights of the network were initialized with weights from a model pretrained on ImageNet.
- It is also common practice to only train the last few fully connected layers of a network and leave the weights for the convolutional layers unchanged from the pre-trained model.

CNNs

- The CNNs for this project were trained with ResNet-18 and AleNet architectures initialized with pretrained weights with the last fully-connected layer with output dimension 1000 replaced by a linear layer with output dimension equal to the number of classes. Also VGG-16 was used with the convolutional layer weights unchanged and only training the fully connected layers.
- The weights that are available for download are tuned for a particular ImageNet classification problem of predicting 1000 different categories, including cars, ships, fish, cats, and dogs.
- Resnet-18 has 20 convolution layers and also 20 batch normalization layers and a single fully connected layer. The first convolution layer utilizes kernals of size 7, but all others use kernals of size 3.
- AlexNet has 5 convolution layers and 3 fully connected layers and no batch normalization layers. AlexNet utilizes kernals of size 11, 5, and 3.
- VGG-16 has 13 convolutional layers and 3 fully-connected layers and no batch normalization layers.

Methods

- Following the method of Rajpurkar, the dataset was split into a training set consisting of 17,978 images (70%), a validation set consisting of 2569 images (10%), and a test set consisting of the remaining 5137 (20%) images.
- All models were initialized with pretrained weights from ImageNet and were trained using a cross-entropy loss function. The VGG-16 models only trained the fully connected layers.
- Parameter update by Adam optimization with an initial learning rate of 0.0001 and which decays by a factor of 0.1 after 5 epochs and again after 8 epochs.
- To reduce overfitting, L2 regularization was used which adds the sum of squares of all weights to the cost.
- A mini-batch size of 32 was selected, and each model was trained for at most10 epochs.

Results

		Accuracy	
Model	*Normalized data*	*2 classes*	*3 classes*
ResNet-18	N	0.864707	0.7078061
ResNet-18	Y	0.8503017	0.6904808
AlexNet	N	0.8483551	0.6778275
AlexNet	Y	0.8302511	0.4512361
VGG-16*	Y	0.8228538	0.6527156
VGG-16*	N	0.7714619	0.6558303

*Pre-trained weights in convolutional layers not trained

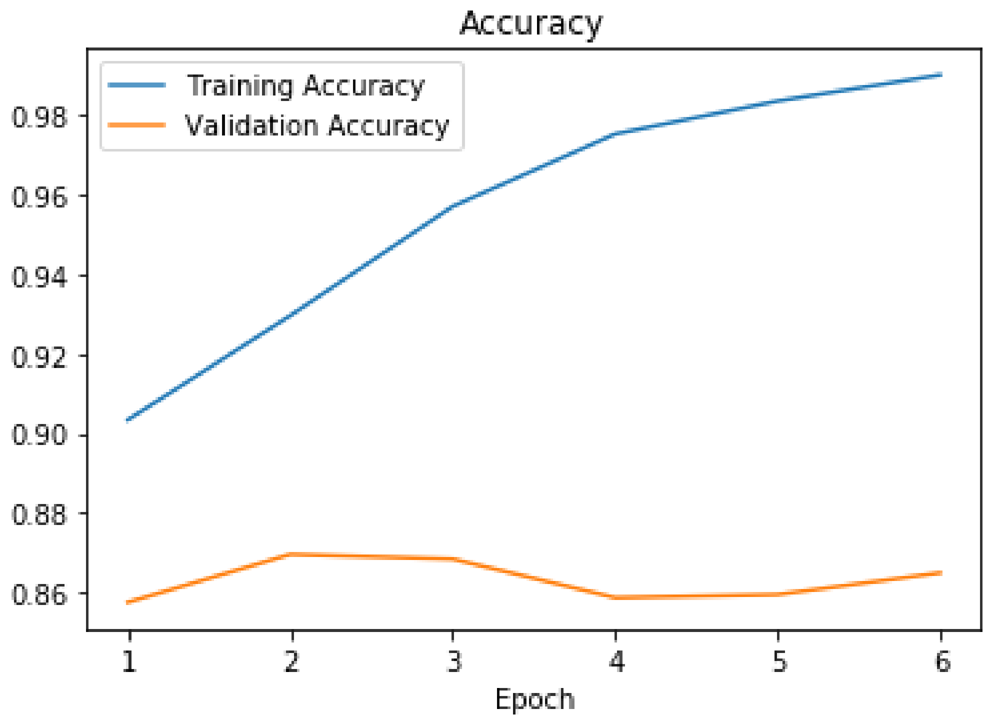
. Each architecture was trained for both the 2-level classification and the 3-level classification and with normalized data and non-normalized data for a total of 12 experiments.

The ResNet-18 model performed better than both AlexNet and VGG-16 for both classifications for both normalized and non-normalized data.

All 3 models showed a decline in prediction accuracy for normalized data compared to non-normalized data.

AlexNet performed better than VGG-16 with both normalized and non-normalized data for the 2-level classification

Results



Model with highest accuracy for predicting normal vs. abnormal images at 86.5%. The training and validation accuracy and loss for this model are shown below. Training was stopped after 6 epochs because training accuracy was greater than 99%.

Confusion matrix

The table below shows the actual and predicted values for the ResNet-18 model which predicted the 3 classes with over 70% accuracy.
Normal images were predicted with 81.38% accuracy and abnormal images without lung opacity were predicted with 76% accuracy.
However images with lung opacity were predicted with only 42.8% accuracy and were misclassified as abnormal without lung opacity 52% of the time.

Table of Actual by Predicted for ResNet-18 based on non-normalized data

Actual	*Predicted*			
	Normal	*Lung Opacity*	*Abnormal: No Lung Opacity*	*Total*
Normal	1403 81.38	20 1.16	301 17.46	1724
Lung Opacity	60 5.46	470 42.81	568 51.73	1098
Abnormal: No Lung Opacity	322 13.91	230 9.94	1763 76.16	2315
Total	1785	720	2632	5137

Conclusions

- This experiment demonstrated that Resnet-18 pretrained on ImageNet can be used to correctly detect abnormal chest x-rays with over 86% accuracy.
- However, discriminating between abnormal images with and without lung opacities proved to be challenging.
- It also demonstrated that ResNet-18 is superior to AlexNet at classifying this set of chest x-rays.
- Although VGG-16 performed the worst at 82% accuracy, this might be due to the fact that the convolutional layer weights were fixed based on the ImageNet training
- These findings support previous findings of the availability of algorithms to automate initial detection of potential pneumonia cases.

References

- Rajpurkar P, Irvin J, Zhu K, Yang B, Mehta H, Duan T, et al. CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. arXiv (cs.CV); 2017.
- J. R. Zech, M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano and E. K. Oermann, Confounding variables can degrade generalization performance of radiological deep learning models, arXi preprint arXiv:1807.00431 (2018).
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet classification with deep convolutional neural networks. In NIPS, pp. 1106–1114, 2012.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR, 2016.
- Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proc. International Conference on Learning Representations http://arxiv.org/abs/1409.1556 (2014).
- https://www.mayoclinic.org/diseases-conditions/pneumonia/symptoms-causes/syc-20354204
- https://www.kaggle.com/c/rsna-pneumonia-detection-challenge
- https://en.wikipedia.org/wiki/Channel_(digital_image)
- http://www.image-net.org/
- https://pytorch.org/docs/stable/torchvision/models.html