**Classfiying Chest Radiographs with CNNs**

**Vinay Bhandaru**

**DATS 6501 Capstone**

**Fall 2018**

**Background**

Pneumonia is an infection that inflames the air sacs in the lungs, causing them to become filled with fluid. Symptoms include cough, fever, chills, and difficulty breathing (1). In the United States in 2015 pneumonia accounted for over half a million visits to the ER and over 50,000 deaths. Diagnosis of pneumonia requires review of a chest radiograph, clinical history, vital signs, and laboratory exams. Pneumonia usually manifests as areas of increased opacity on a chest radiograph. Chest X-rays are the most common imaging examination tool used in practice.

In an effort to improve diagnostic services the Radiological Society of North America (RSNA) in collaboration with the NIH, The Society of Thoracic Radiology, and MD.ai created a database of over 25,000 labeled chest radiographs. RSNA is an international society of over 54,000 radiologists, medical physicists, and other medical professionals across the world. The data was provided to Kaggle’s machine learning community because of the potential for ML to automate initial detection of potential pneumonia cases in order to prioritize and expedite their review.

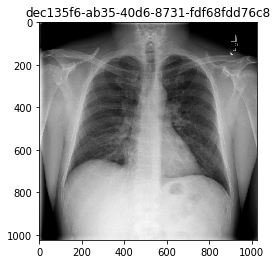
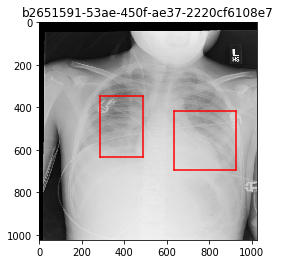
The objective of this project is to determine how well chest radiograms can be classified as normal or abnormal using convolutional neural networks and to compare the performance of several popular CNN architectures.

**Materials and Methods**

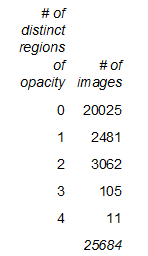
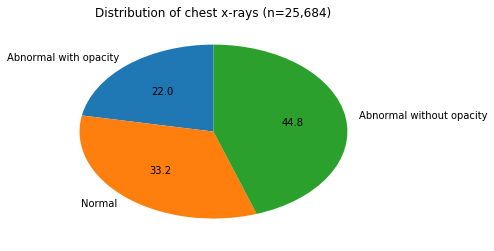
Description of Data

The images and label data were available on Kaggle.com for the RSNA challenge. Kaggle is an online community of data scientists and machine learners owned by Google, Inc. There were 25,684 chest radiograph images available in dicom format (.dcm). A python library to read the radiograph image files and extract the pixel array is pydicom. The filename for each chest radiograph image is a unique 36-character patient identifier which links the image to its label in a separate datafile. Image files range from 32 kb to 362 kb, and the average size is 130 kb. All files combined are 3.2 GB. Each full size raw image is a grayscale image of 1024 x 1024 pixels. A grayscale image has only one channel for each pixel. Brightness intensities at each pixel range from 0 to 255. By contrast, a standard digital camera image a red, green, and blue channel for each pixel, so that each pixel is a combination of primary colors with contributions indicated by brightness intensities ranging from 0 to 255.

Each chest radiograph is classified as either 1) Normal, 2) No Lung Opacity / Not Normal, or 3) Lung Opacity. Image class is provided in a separate file for each patient identifier. An additional file was available with information about locations within an image of lung opacities. The file specifies a minimum vertical and horizontal coordinate along with width and height from which bounding boxes can be constructed (see image below). An example of each type of image is provided below. The image on the far left is normal, the image in the middle is No lung opacity / not normal, and the image on the far right is lung opacity and the red bounding boxes indicate the regions of opacity.

Out of the 25,684 images 8525 (33.2%) are classified as Normal, 11,500 (44.7%) are classified as No Lung Opacity / Not Normal, and the remaining 5659 (22.0%) are classified as Lung Opacity. In images classified as Lung Opacity, the number of bounding boxes ranges from 1 to 4 as indicated in the table below.



CNNs

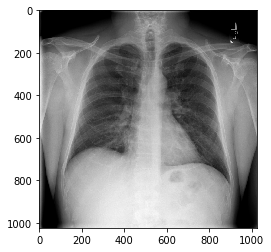
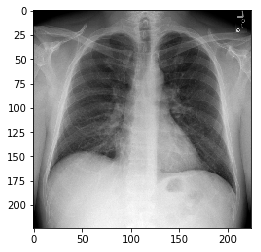
Convolutional neural networks have become popular for image recognition in recent years. One reason for their popularity is that they make better use of spatial information in an image through the use of convolution filters, which is lost in a fully connected layer or multi-layer perceptron. Many different CNN architectures have been implemented, including variations of ResNet, DenseNet, 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. CheXNet is an algorithm designed to detect pneumonia from chest x-rays using DenseNet-121, which is a 121-layer CNN. 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. Then the network was trained on 100,000 chest X-ray images with 14 diseases. A nearly identical approach was used by Zech in a study which focused on training deep learning models to predict pneumonia from chest x-rays. One difference is that an additional dense layer was added to the network instead of replacing the final fully connected layer.

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. Both of these approaches will be explored in this experiment.

Pre-processing

As in both Rajpurkar's and Zech's studies, all images were resized to 224X224 and converted from one color channel to 3 color channels. Each image is 224X224X3, which is 150,528 inputs per image. This was accomplished by cropping the edges of each image to yield an image which is 896 x 896 and then sampling every fourth pixel, resulting in an image which is 224 x 224. The image on the left is the original 1024 x 1024 image and the image on the right is the cropped and downsampled 224 x 224 image. The images were then 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] which follows the approach of Rajpurkar.

Model training

Pytorch and torchvision in Python were used for all model training utilizing GPU wherever possible. The torchvision package and specifically the models subpackage contain definitions for many common model architectures, such as AlexNet, different versions of VGG, and different versions of ResNet. Models can either be initialized with random weights or weights from training the model on ImageNet can be downloaded. 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. Most of these pre-trained models assume an image size of 224X224 with 3 color channels.

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. The CNNs for this project were trained with ResNet-18 and AleNet architectures with the last fully-connected layer with output dimension 1000 replaced by a linear layer with output dimension equal to the number of classes. The images can either be classified into 2 or 3 classes, depending on whether the abnormal images with lung opacities are separated from the abnormal images without lung opacities.

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 with parameter update by Adam optimization. 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 10 epochs. The learning rate started out at 0.0001 and at epochs 5 and 8 decreased by a factor of 0.1.

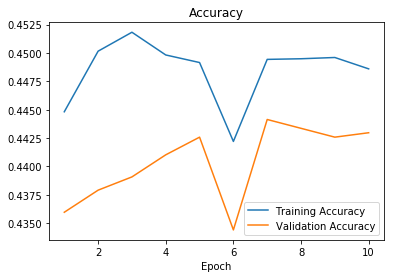
**Results:**

The VGG-16 model could not be trained because of a memory error. To rectify this problem, the weights in the convolutional layer were held fixed so that only the weights in the fully connected layers were updated during training. The results of training each of the 3 CNN architectures are shown below. 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.

|  | | *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 | | | |

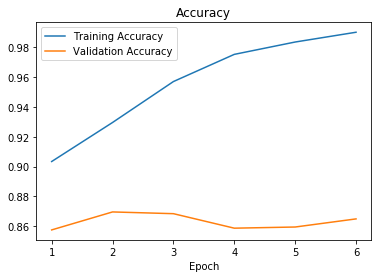
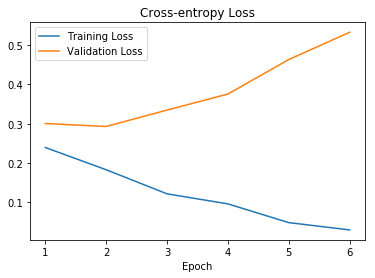
The ResNet-18 model performed better than both AlexNet and VGG-16 for both classifications for both normalized and non-normalized data. All 3 models showed a decline in prediction accuracy for normalized data compared to non-normalized data. AlexNet performed better than VGG-16 with both normalized and non-normalized data for the 2-level classification. However, for the 3-level classification it performed better than VGG-16 for non-normalized data but much worse for normalized data. The training and validation accuracy for each epoch are plotted below, indicating that the model did not train well with standardized data.

**Training and Validation Accuracy for AlexNet with 3-level classification and normalized data**



Every CNN architecture predicted Normal vs. abnormal images more accurately than predicting the 3-level classification. ResNet-18 performed the best at predicting the 3-level classification in the test set at 70.8% for non-normalized data and 69% for normalized data. It also has the 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%. The validation accuracy nearly reached its peak value after only 1 epoch, probably because of the use of pretrained weights.

**Training and Validation Cross-entropy loss and accuracy for ResNet-18 with 2-level classification of non-normalized data**



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 since training the entire VGG-16 model exceeded the computational capacity of the computer. With greater computational resources, VGG-16 might have outperformed both AlexNet and ResNet-18. These findings are promising in terms of developing an algorithm 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>