Assessing Ventilator-Associated Pneumonia (VAP) Using Deep Learning Methods

Computer Integrated Surgery II, Spring 2019

Group 13: Suraj Shah
Mentors: Dr. Mathias Unberath, Dr. Jim Fackler, and Dr. Jules Bergmann
May 9th, 2019

Background

Pneumonia continues to be a disease that has long-lasting and detrimental effects on the U.S. healthcare system. Nearly one million diagnoses are made per year, with more than 50,000 individuals dying from the disease each year[1]. Chest X-rays are the empirical standard for diagnosing pneumonia, however, the diagnosis for each individual patient is still heavily reliant on the knowledge and practice of skilled radiologists. Furthermore, complications are much more exacerbated when patients are in critical-care situations such as the ICU (Intensive Care Unit) or PICU (Pediatric Intensive Care Unit). Mechanical ventilation (and intubation) is a crucial, life-sustaining ICU therapy – with more than half of all ICU patients receiving ventilated within the first 24 hours of ICU admission[2]. However, ventilation greatly increases the risk for acquiring pneumonia, known as VAP (Ventilator Associated Pneumonia). VAP specifically refers to "pneumonia developing in a mechanically ventilated patient more than 48 hours after tracheal intubation."[3] The reason ventilation induces pneumonia is because the body is at a fragile state and susceptible to diseases, including bacterial infections that are attributed to VAP. There are other risks involved as well, as further disease progression, volume overload, latrogenic infection, and ventilator injury. VAP has had such a deteriorating effect in the ICU ward that it is now the leading cause of mortality among nosocomial infections and the leading cause of nosocomial morbidity.[3] To further highlight the problems associated here, 10-20% of ICU patients are diagnosed with VAP annually and acquiring VAP increases the risk of mortality by 30%. Ventilator-associated complications are also correlated with a much greater length of stay and time under ventilation. This leads to greater strain on the entire healthcare value chain, from the provider, insurer, and most importantly, the patient.

Problem and Goal

While there is a multi-disciplinary team working on addressing the issues arising from VAP in the ICU (including the PICU team working on identifying biomarkers and the ID team working on appropriate cultures and antibiotics), there hasn't been a comprehensive study connecting the radiology component of monitoring patients with VAP and/or risk of VAP. This is mainly due to the clinical data collection process. The X-ray images that are collected in the ICU and PICU occur over many different hospitals, at different orientations of patients, on different machines, and either at inspiry or expiry. There is not a standardized process for data collection; leading to an aggregation of thousands of X-ray images, but ones that are not easily ingestible into an algorithm that is able to readily classify the risk of VAP for a specific patient. It is imperative to build a system that is able to detect pneumonia for ventilated patients in the ICU that does not solely rely on radiologist diagnosis. My project aims to prepare an automatic classifier to be used in the ICU to diagnose VAP.

Technical Approach

Data Aggregation and Screening

To ingest into the pipeline to train my models, I first had to procure access to public available chest X-ray data. The majority of work here encompassed training on adult patients, given what was available to the public. We initially hoped to have access to pediatric chest X-ray data; however, given the IRB approval timeline, and the sensitivity regarding pediatric data, this was not accomplished in the duration of this course. Regardless, this pediatric data from the John Hopkins Hospital PICU would only have been available to test the models. For actual training, there wouldn't be enough data with a critical mass of patients to train an accurate model that could be used for general-purpose classification. Thus, we were limited to training on the two largest publicly available datasets, MIMIC-CXR[5,6] and NIH ChestX-ray14[6]. Below we see how the data is organized by the two datasets.

- MIMIC-CXR: 371,920 chest x-rays associated with 227,943 imaging studies. Chest X-rays were taken from patients admitted to Beth Israel Deaconess Medical Center between 2011 and 2016. Published by the MIT Laboratory for Computational Physiology, following upon their work with MIMIC-III, which contains critical care patient data from more than 40,000 intensive care unit stays. Unfortunately, the two databases are not linked, but the lab hopes to bridge them in the future, which would help with analysis, given that researchers would have access to both ICU clinical data and imaging data.
- NIH/Stanford: Chest-Xray14, which contains 112,120 frontal images of 30,805 unique patients. Released by the NIH, these images were collected over 15 years in numerous studies at Stanford Hospital. Similar to MIMIC-CXR, this contains a framework that labels each image with up to 14 unique thoracic pathologies.

Both datasets gave access to frontal and lateral images. However, for training a convolutional neural network, it is important to keep standardization with the orientation of the image. Thus, for data ingestion, I only used the frontal-view images, given that there are much more of than (almost a 3:1 ratio), which makes for greater training sizes.

Given that, on average, multiple studies were collected on each patient, it was imperative to keep the training, validation, and test data separate with no patient overlap. Given the scope of the project, I picked appropriate sizes for each dataset (starting with an initial training size of 2700 images and scaling up to 65,000 images for both MIMICS and NIH). A validation dataset was provided by MIMICS but not NIH; I had separated the validation datasets using cross-fold validation into sizes of 10,000 images. The dataset sizes were chosen based off various sizes found in previous literature.

Lastly, I had to ensure the data was appropriately sized to ingest into the model. Given that each image was held at slightly different orientations (given the pose of the radiologist taking the x-ray), encompassed different scales of adult bodies, and so forth, resizing was needed. Using standard normalization techniques, the images were downscaled the images to 224×224 and normalized based on the mean and standard deviation of images in the ImageNet training set (which is what many of the pre-trained models learned on). The training data was also incorporated with random horizontal flipping to introduce more variability into the ingestion pipeline and in hopes of training the model to be more robust to feature detection.

Architecture and Testing Approach

I tested 3 different approaches on 3 pre-trained Convolutional Neural Networks: VGG, ResNet, and DenseNet, which are all detailed below. Training was conducted on a small dataset from MIMICS (25,000 images) for all 3 models, and the best performing model was further trained on the actual training dataset which consisted of 65,000 images. A smaller dataset was chosen to verify the performance of all 3 models given the time it took to train the full dataset (roughly 12 hours), even with utilizing the GPU structure on MARCC (Maryland Advanced Research Computing Center).

1. VGG-16[8]: developed in 2014, this network consists of 16 layers, and is appealing because of the uniform architecture. In terms of performance, it is one of the most preferred choices in extracting features from images, which is very relevant when pursuing relevant factors in the chest X-ray images. The VGG convolutional layers are followed by 3 fully connected layers, which double in width after each pooling layer. One concern with VGG is that has 138 million parameters, which could be a potential hurdle in terms of difficulty to handle.
2. ResNet[9]: also known as Residual Neural Network, this was developed in 2015 and focuses heavily on "skip connections" (gated recurrent units used in RNNs) and batch normalization. The idea is to create a direct path between the input and output to the network implying an identity mapping, as well an added layer that learns the features on top of already available input. The architecture consists of 152 layers but is much less parameter dependent, with nearly 1/6 of the parameters of VGG, thus reducing the complexity involved.
3. DenseNet[10]: an extension of ResNet, also known as Densely Connected Neural Network. DenseNet's methodology proposes concatenating outputs from the previous layers instead of using the summation (ResNet merges previous layers with future leaders).

After initial training of all three models, modified ResNet was shown to be the best-performing network for binary classification, with a test accuracy approaching 94%. VGG lagged behind, with accuracy approaching 91%, and DenseNet with accuracy approaching 92%. X also had the highest AUC (Area Under the Curve) from ROC (Receiver Operating Characteristic), with an AUC of .8193.

With the ResNet model, I replaced the final fully connected layer with one that has a single output, after which I applied a probabilistic softmax output function to give the binary prediction.

The weights of the network are initialized from parameters learned from initialized training on ImageNet, the world's largest and most robust image classification dataset (1M+ images, 1000 classes). Given the binary structure here, I also updated the weights of the model to optimize with binary cross entropy loss. The model is optimized end-to-end using Stochastic Gradient Descent with a learning rate of 0.001 and momentum of 0.9. The gamma for the learning rate (decay factor) is 0.1, i.e. the learning rate is reduced 10x if validation loss plateaus. My final model is implemented and tested with the epoch that represents the lowest validation loss.

Beyond binary classification, an alternative approach to classification was to distinguish between "No Finding," "Pneumonia," and "Other Thoracic Diseases." The purpose here was to understand if there was a significant difference between pneumonia and other thoracic pathologies, and if so, could a CNN be trained to distinguish between these three categories. There is a caveat, however, given that a patient diagnosed with pneumonia could also be diagnosed with other thoracic pathologies. In terms of ground-truth, I labeled these patients in the Pneumonia only category, given my prediction that pneumonia had much stronger features in terms of extraction for the CNN. Results are shown in the results section.

Finally, I utilized a Class Activation Mapping technique (CAM) developed by CSAIL at MIT[11] to feed forward in the most important classification features used by the model when developing the prediction for pneumonia classification. This is a color-coded heatmap that highlights in red where the model places highest importance for its prediction. While this can be applied to any pre-trained network, the beauty here is that I can take the best-performing trained network on pneumonia detection and apply it specifically to the feature extraction task at hand. In this project, I used the modified ResNet to visualize the heatmap. For this specific CAM, the algorithm projects the weights of the last convolutional layer onto the feature maps that are produced by the layer, using the weighted sums to determine the highlighted regions. The maps are scaled to the dimensions of the image with the correct prediction.

To test against clinician detection of the most important features, I compared the CAMs of images to marked areas of the same images by one of my clinical mentors, Dr. Jules Bergmann. It is important to note that he is not a radiologist and focuses mainly on pediatric patients; that being said, his predictions are valuable nonetheless given the breadth of work he covers in VAP detection in the PICU. A few examples of comparisons are shown in Results.

Results

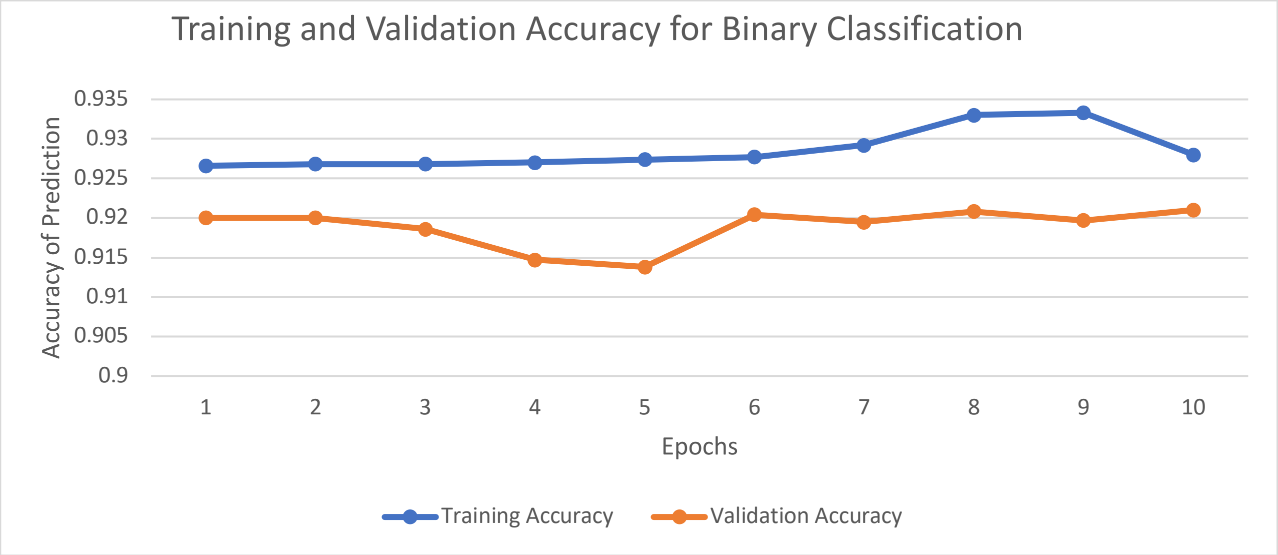


Figure 1: Training and Validation Accuracy on Pneumonia Detection

The above chart shows the training and validation accuracies of my modified ResNet, the best-performing model over 10 epochs on a training set of 65,000 images and a validation set of 10,000 images. Overall run time took slightly longer than 12 hours. It is important to note that both training and validation accuracies gradually plateau towards the end of the epochs. It could have been interesting to see if this accuracy plateau would have occurred with a sample of say 100 epochs, but this training would have lasted much more extensively than we had bandwidth for.

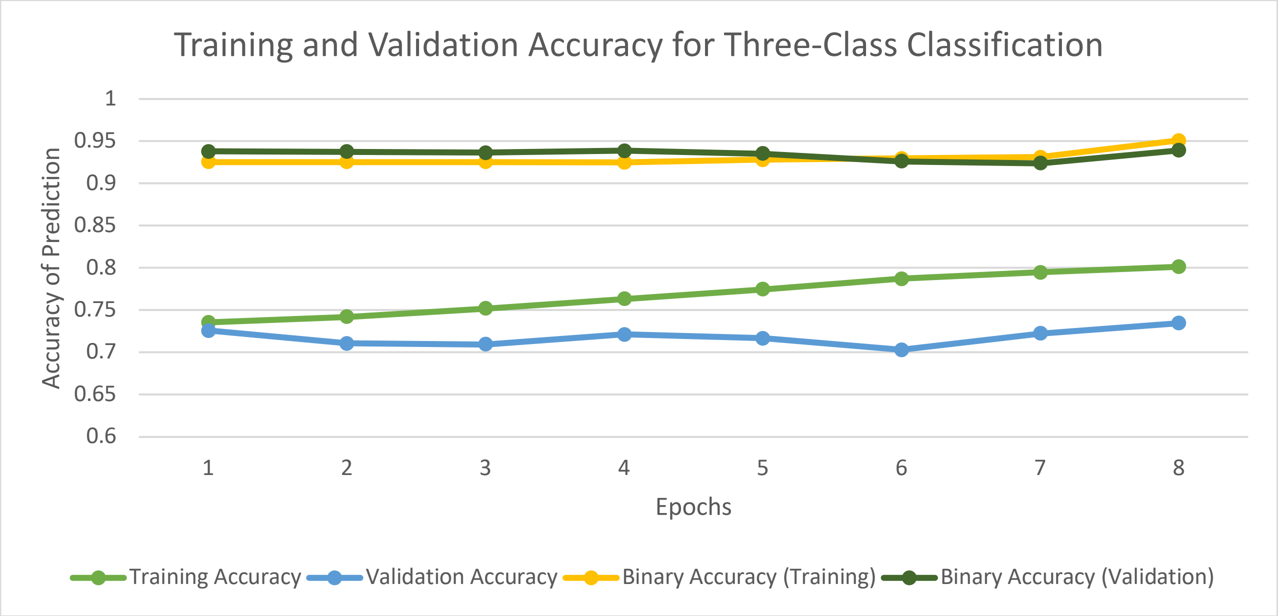


Figure 2: Training and Validation Accuracy for Three-Class Classification

Figure 2 displays two important trends. First, the model performs much worse when classifying the images into three separate object classes vs. binary detection of pneumonia. This is interesting yet predictable given that the introduction of a new class, Other Thoracic Pathologies, is much more so of an arbitrary class rather than a fully distinguishable class in and of itself. To see whether this is a problem of the model itself or rather my own classification methods, I grouped the predictions of Other Thoracic Pathologies and No Finding into a single class in post-training analysis, essentially creating the same two classes as before. When comparing the accuracies of these two classes, the second important trend emerged. Accuracy shot up to levels similar to pneumonia detection in Figure 1. Thus, we can conclude that the model can predict pneumonia with high accuracy but suffers when separating a healthy patient from the bucket of other thoracic pathologies. Possible next steps would be for the model, instead of using a softmax to output one classification, outputs a vector of classifications, for each of the thoracic pathologies in the dataset.

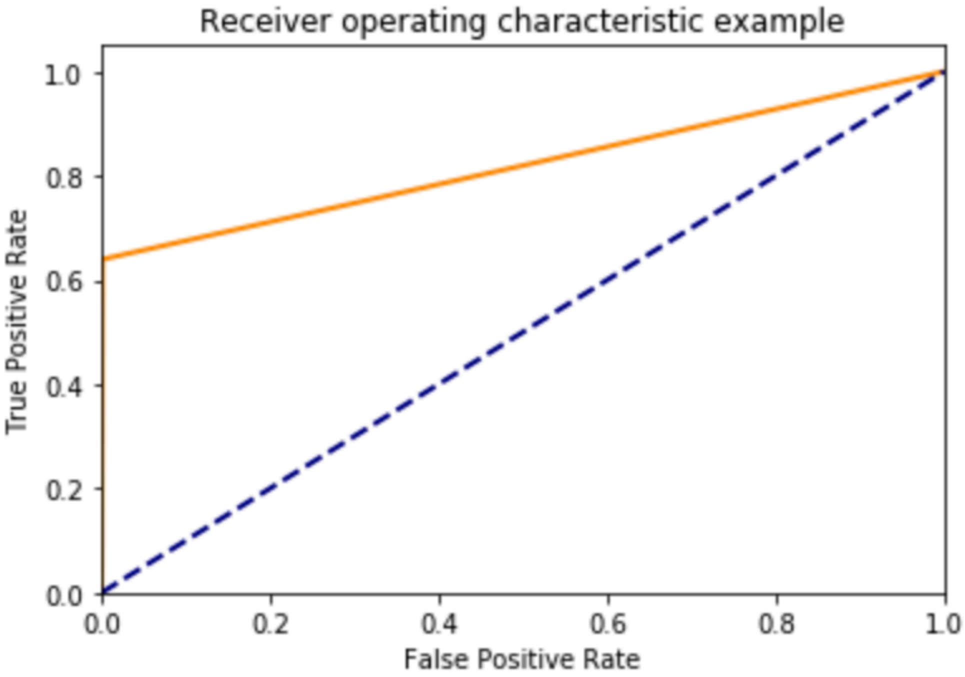
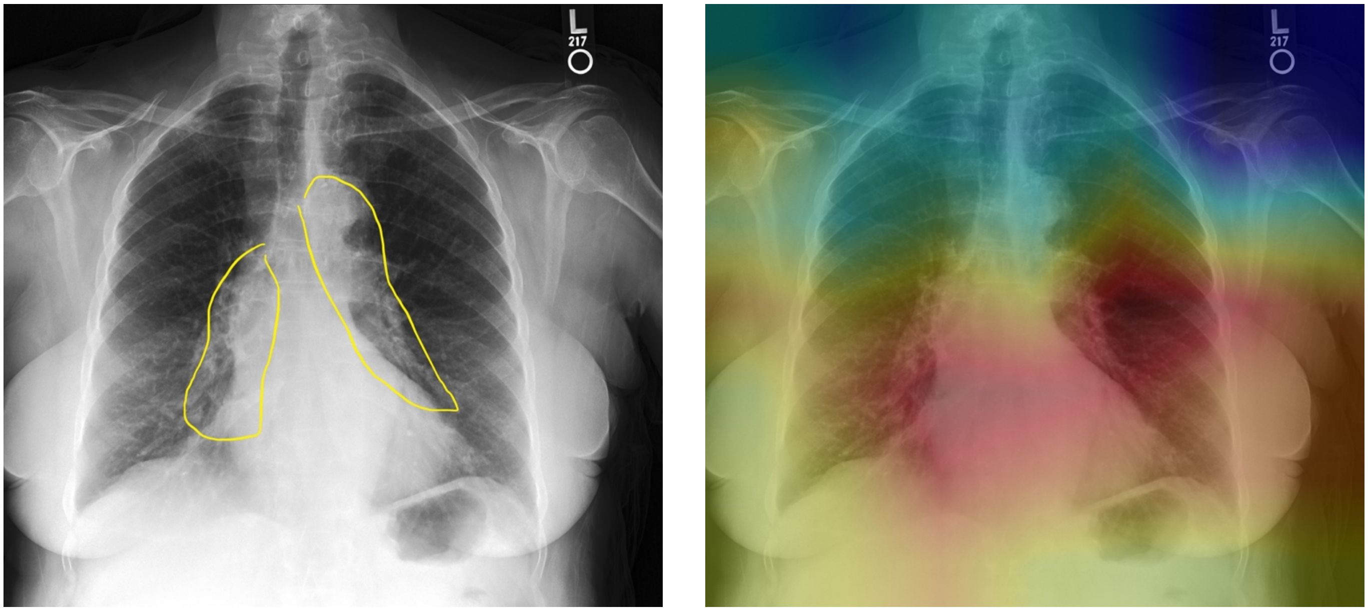
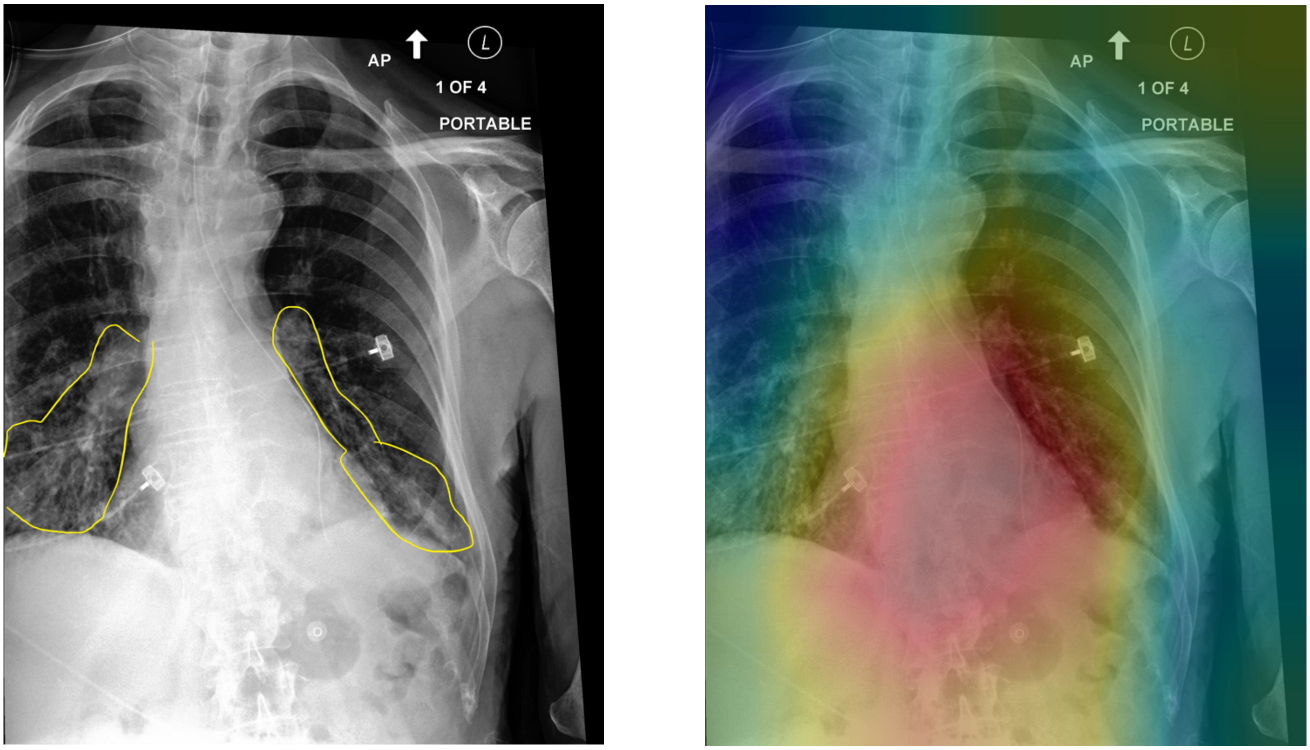


Figure 3: ROC Curve for best-performing model (ResNet) on test data

With many classification problems in data science, it is important to measure performance beyond simple accuracy. Thus, I mapped a ROC curve (Receiver Operating Characteristics) and measured the Area Under the Curve (AUC). Given that my task for classification is for disease detection, the importance of ROC is much more profound, given the likelihood of false positives and negatives. The AUC represents the degree of separability – the larger AUC, the better the model is at predicting truths verse false positives and negatives. The AUC is: .8193, which is similar to previous works that perform pneumonia detection.



Figures 5 & 6: Comparison of physician feature detection vs. model CAM on patient with pneumonia



Figures 7 & 8: Another example of physician vs model feature extraction. Here, notice how the model picks up on the left lung nodules while the physician is concerned with both left and right lungs.

Significance & Conclusion

Pneumonia continues to be a significant factor in patient mortality, both in critical-care settings and out-patient settings. Especially with ventilator-assisted patients, the risk of mortality, length of hospital stays, and further complications significantly increases with pneumonia contraction. Given the lack of trained radiologists in such settings, it is imperative that the medical community can build an automatic classifier that can augment physicians in these settings to truly help diagnose pneumonia at an early stage. Through my work in CIS II, I was able to build the foundation for an automatic classifier that can detect pneumonia from frontal view chest X-rays, with complementing heatmaps that can help visualize areas of concerns, especially to physicians who may not be trained in radiology. My repurposed model performs the pneumonia detection with accuracies mirroring those of practicing radiologists, but further work is needed to validate this score with more robust scoring techniques such as the F1 method (calculating a mean of precision and recall). However, with initial AUCROC analysis, we can conclude that the classifier is progressing towards fairly accurate pneumonia diagnosis. I hope this technology can be implemented and trained in the future on both ventilated and pediatric patients, areas of concern for our clinical collaborators. Furthermore, this could potentially be revolutionary for low-resource settings with limited access to medical imaging experts.

Accomplished vs. Planned

Below are my deliverables, as stated in the project plan of the course. These were subject to revision after my project plan given two unforeseen circumstances: 1) former partner had to drop the course, 2) the chest x-ray data wasn't presented as time-series data, which made accomplishing many of the goals much more difficult if not possible throughout the course of the semester. However, I was successful in completing my Expected Deliverables for the course as shown.

Minimum:

- A segmented database of X-ray cohorts
- Trained algorithm (pytorch model) for static image prediction

Expected

- Trained algorithm (pytorch model) for 3-class classification
- Trained saliency mapping algorithm for physician use

Maximum

- A trained pytorch model file for unsupervised learning of visual features
- Clinically actionable report of F1 scoring compared against current radiologists

Future Work

While I would love to continue work on building out this model in the future, I will be graduating in May and starting a full-time job a few months after graduation. However, I will outline the next steps for the project for discussion and for whomever wishes to continue this work.

Computationally, there is room for interesting analysis to be conducted on top of the output of the CNN. The models I used in this project were trained on ImageNet, which contains 1 million images to cover most types of classification (up to 1000 labels). To increase performance, one could feasibly increase the size of the dataset by a factor of 10-100x, but that would concern much more manual annotation, placing a burden on human effort and is much more extensive than what is currently available in the data science community. Thus, is the burden of supervised learning. However, many researchers, including the Facebook AI Research Group[12], have developed a set of techniques that apply clustering of the features that are outputted by the final weights of the convolutional layer of the network, which are clustered into 'pseudo labels.' This deep clustering method could prove for much more interesting feature extraction as compared to the binary classification that currently exists with my model, specifically with the generalization of pneumonia features across demographics, presence of equipment such as ventilations, and imaging equipment.

Experimentally, I was unable to test my model on pediatric patient images in this course due to the approval timeline. Most, if not all, of the available chest x-ray data is on adult patients, including the data that my model trained on. However, given that the body structure and lung relative placement is much different for pediatric patients, it would be necessary to see if my model is also robust in detecting pneumonia for these patients. Additionally, given that VAP was the goal of this project, it would be necessary to include pediatric patients who are ventilated, to truly determine if ventilation poses any difficulty in pneumonia detection as well.

What I Learned

This project gave me an incredible opportunity to understand deep learning as applied to computer vision techniques and handling the curation of large-scale internet databases. I worked for the first time on a supercomputer environment (MARCC), built my own data repository, and produced a trained model. Before this project, I was unaware that pre-trained networks have useful classification features and optimized as such to the point where one can manipulate the networks for one's own research. My previous experience had been building a bare-bones model, but it was much more efficacious to tailor an elegant solution to the problem at hand. Furthermore, even given the prevalence of such pre-trained networks, data ingestion and cleaning continue to be a problem for data scientists, even within the imaging community. Even more lacking is the linking of images across time (time-series), which severely undermines the type of analysis one can run. A working improvement for those releasing public datasets is to include time-characteristics where applicable. Lastly, I realized that pneumonia detection, while heavily dependent on the chest x-ray imaging within the scope of my project, accounts for much more variables than we had access to, given the format of the data. Once MIMIC is able to link

MIMIC-CXR and MIMICS-III, much more interesting analysis, that combines imaging with the clinical log of a patient, can be used in pneumonia detection.

Technical Appendix

All relevant scripts (scratch and scripts in progress are not shown) are hosted for documentation purposes on a public GitHub link that is posted on my project website. Below is a brief description of each of the scripts and their purposes. The data, for HIPAA-compliance, data storage, and confidentiality reasons, is not hosted on the GitHub. It is hosted through Dr. Unberath's account on the MARCC server and can be accessed through request. Also, the trained PyTorch model files (the output of training sessions) were too large to be saved on the GitHub repository, so are also hosted on the MARCC server as well. Each of the scripts explains how to run and how to format the inputs accordingly.

- **ClassActivationMapping.py**: for the best-performing model (ResNet), takes in an image input and performs prediction and outputs a class activation map, visualizing the highest weighted regions of the image as calculated by the model.
- **DenseNet.py**: reformats the pre-trained DenseNet model to apply to the dataset, modifies parameters and makes optimization choices, and outputs both accuracy and ROC plots for a given number of training epochs.
- **ReroutingFiles3Class.py**: reformats and cleans a set number of training and validation data into three classes (No Finding, Other Thoracic Pathologies, and Pneumonia) for seamless ingestion into the Pytorch models.
- **ReRoutingFilesBinary.py**: reformats and cleans a set number of training and validation data into the binary classes (Pneumonia vs. No Pneumonia) for seamless ingestion into the Pytorch models.
- **ResNet.py**: reformats the pre-trained ResNet model to apply to the dataset, modifies parameter and makes optimization choices, and outputs both accuracy and ROC plots for a given number of training epochs.
- **ResNet3Class.py**: applies the ResNet model to the 3-class classification problem.
- **ResNetLargeDataset.py**: applies the ResNet model to the largest dataset curated, modifies parameter and makes optimization choices, and outputs both accuracy and ROC plots for a given number of training epochs.
- **VGG.py**: reformats the pre-trained VGG model to apply to the dataset, modifies parameter and makes optimization choices, and outputs both accuracy and ROC plots for a given number of training epochs.
- **testResNet.py**: applies a trained ResNet model to a select group of testing data to verify accuracy on data that has not been seen by the model in training & validation.

References

1. "Pneumonia Can Be Prevented-Vaccines Can Help | CDC." Centers for Disease Control and Prevention. 2018. Centers for Disease Control and Prevention. 20 Apr. 2019 &lt;https://www.cdc.gov/pneumonia/prevention.html?CDC_AA_refVal=https%3A%2F%2Fwww.cdc.gov%2Ffeatures%2Fpneumonia%2Findex.html&gt;.
2. Kollef, M. H., Dr. (2005). What Is Ventilator-Associated Pneumonia And Why Is It Important? Respiratory Care, 50(6), 714-724. Retrieved February 26, 2019
3. Kirton, Orlando. "Mechanical Ventilation - The American Association for the Surgery of Trauma." American Association for the Surgery of Trauma, AAST, 2011. 20 Apr. 2019. www.aast.org/GeneralInformation/mechanicalventilation.aspx.
4. He, K., Zhang, X., Ren, S. and Sun, J. (2019). Identity Mappings in Deep Residual Networks. [online] arXiv.org. Available at: https://arxiv.org/abs/1603.05027 [Accessed 18 Feb. 2019].
5. Johnson AEW, Pollard TJ, Berkowitz S, Greenbaum NR, Lungren MP, Deng C-Y, Mark RG, Horng S. MIMIC-CXR: A large publicly available database of labeled chest radiographs. arXiv (2019).
6. Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng C-K, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation 101(23):e215-e220 [Circulation Electronic Pages; http://circ.ahajournals.org/content/101/23/e215.full]; 2000 (June 13).
7. Wang X, Peng Y, Lu L, Lu Z, Bagheri M, Summers RM. ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases. IEEE CVPR 2017, http://openaccess.thecvf.com/content_cvpr_2017/papers/Wang_ChestX-ray8_Hospital-Scale_Chest_CVPR_2017_paper.pdf (link is external)
8. Karen Simonyan, Andrew Zisserman. "Very Deep Convolutional Networks For Large-Scale Image Recognition." arXiv:1409.1556v6 [cs.CV] 10 Apr 2015
9. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. "Deep Residual Learning for Image Recognition."   arXiv:1512.03385 [cs.CV] 10 Dec 2015
10. Huang, Gao and Liu, Zhuang and van der Maaten, Laurens and Weinberger, Kilian. "Densely connected convolutional networks." arXiv:1608.06993 [cs.CV]. Proc. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)
11. Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, Antonio Torralba. "Learning Deep Features for Discriminative Localization." Computer Science and Artificial Intelligence Laboratory, MIT. Proc. CVPR (2016)
12. Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. "Deep Clustering for Unsupervised Learning of Visual Features." arXiv:1807.05520 [cs.CV]. Proc. ECCV (2018).\