View-Specific Models For Chest Radiograph Interpretation

Miguel Aguilar
Stanford University
Department of Electrical Engineering
miguel6@stanford.edu

Stephen Lopez
Stanford University
Department of Electrical Engineering
slopez27@stanford.edu

Abstract

*Chest radiography is an important tool used for examination, screening, diagnosis, and management of many life-threatening diseases. Automated chest radiograph interpretation will allow for an increase in work throughput and potentially increase correct diagnoses and decrease false detection. Chest radiography utilizes three views (posteroanterior, anteroposterior, and lateral views) which are typically interpreted differently. Irvin et al. [5] from the Stanford Machine Learning Group present CheXpert, (**Che**st e**Xpert**), a large dataset of chest X-rays and an open competition for automated chest x-ray interpretation released by the Stanford Machine Learning Group. Current models for automated chest radiograph interpretation train a single model for all three views and fail to acknowledge that the views are interpreted differently. This project will be proposing view-specific models that will be trained on different view types with the intention of determining whether or not the view-specific models perform better than the single model trained on all views.* ***Need to add 1-2 sentence summary for all sections***

1. Introduction

Chest radiography is the most common imaging examination globally, critical for screening, diagnosis, and management of many life threatening diseases. Automated chest radiograph interpretation at the level of practicing radiologists could provide substantial benefit in many medical settings, from improved workflow prioritization, clinical decision support to large-scale screening and global population health initiatives, and increased workflow throughput as these are problems still needing a solution in the medical field [1]. CheXpert is a large dataset that consists of 224,316 chest radiographs labeled for the presence of 14 common chest radiograph observations from free-text radiology reports and captures uncertainties present in the reports by using an uncertainty label. Each chest radiograph is done with one of three views, posteroanterior, anteroposterior, or lateral, each of which interpreted differently in radiology practice. However, current models for automated chest radiograph interpretation are trained on all views, and fail to take into account the fact that the different views are interpreted differently. CheXpert can be used to predict the probability of presence of 14 common radiographic observations, "No Finding", "Enlarged Cardiomediastinum", "Cardiomegaly", "Lung Lesion", "Airspace Opacity", "Edema", "Consolidation", "Pneumonia", "Atelectasis", "Pneumothorax", "Pleural Effusion", "Pleural Other", "Fracture", "Support Devices." This paper is looking to develop view-specific models for each view, both via complete training and via transfer learning to analyze the impact that view-specific models can have on interpreting chest radiographs.

Our model will be taking a chest radiograph of a specific view, posteroanterior, anteroposterior, or lateral, and then it will be outputting a vector denoting the presence of the 5 available observations, Atelectasis, Cardiomegaly, Consolidation, Edema, and Pleural Effusion, similar to the original ChexPert paper [5], shown in Figure 1.

2. Related Works

There is a large body of work that attempts to address the problem of automated interpretation of chest radiographs [10][3]. Most notably is the CheXpert paper [5], that introduces the CheXpert dataset as well as introduces a baseline of its own. They narrowed the scope of the problem to just detecting the presence of 5 chest radiograph observations rather than the 14. This paper developed and trained their model on all views of the chest radiographs, failing to take into account that each view is interpreted differently for different observations. They proposed a baseline model utilizing a DenseNet121 [4] architecture and a cross-entropy loss function. Additionally, CheXpert presented a way to deal with the uncertainty , u, labels in the dataset that will be explained in Section 4. However, the key drawback to the model presented by CheXpert was that the model was trained on all three views.

Another attempt at this problem was a project [2]

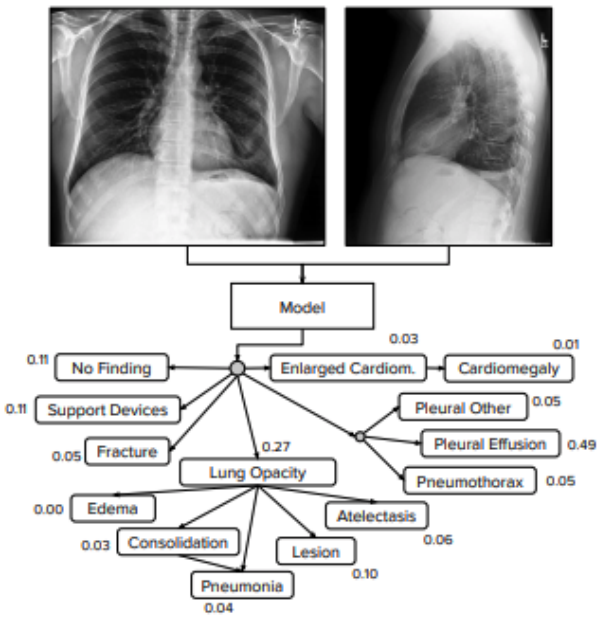

Figure 1. Concrete example of the input-output behavior the model [5]

that wanted to replicate CheXpert's results with a Naive DenseNet model. This model utilized the same approach that CheXpert used when dealing with the u labels in the dataset. This model's drawback was also that it was trained on all three views, rather than each one individually.

Since our approach involved developing an individual model for all three views, we wanted to investigate transfer learning and its applications to medical imaging. Studies have shown that transfer learning can be extremely helpful in reducing the time needed to train models as well as in achieving higher performance metrics [8][7][6].

Additionally, medical imaging has issues with class imbalance [9] that was attempted to be solved by [6] through a new loss function that we explored using. Ultimately, we found that developing the understanding of the new loss function to a degree that we felt comfortable with in order to apply it to this multi-label classification problem instead of the original segmentation problem it was designed to solve was taking up a lot of time, so we chose to not use it in our approach.

3. Model

For our preliminary model, we implemented an architecture that was defined as such: [Conv $\rightarrow$ ReLU $\rightarrow$ MaxPool] $\rightarrow$ FC. For our final model, we initially planned on implementing a number of different architectures that were mentioned in the CheXpert paper [5], such as DenseNet, ResNet, and ResNeXt, however due to time constraints, we first tried AlexNet, which we chose due to its simplicity and quick training, but ultimately decided to focus on DenseNet121 [4] since it performed better. There were some adjustments that we had to make to the original DenseNet model priot to using it. The first change was that we needed to change its classifier layer to simply output 5 values rather than 100 since our model was classifying the presence of 5 chest radiological observations. The second change that we needed to make was to its first feature layer such that we could simply input a single grayscaled image rather than a 3-channel duplicate of the original chest radiograph. First and foremost, we trained our model on the anteroposterior dataset since that was our largest model, and the model was trained for 15 epochs. Following this, we used the model's final state and fine tuned it to better accommodate the smaller lateral and posteroanterior datasets.

All CNN models were trained on NVIDIA V100 GPUs. The data was split into a $75/20/5$ Training/Validation/Testing split. The optimization algorithms that we experimented with were Stochastic Gradient Descent:

$$w = w - \eta \nabla_w L_i$$

and one of its variants, Adaptive Moment Estimation (Adam):

$$w^{(t+1)} = w^{(t)} - \eta \frac{\hat{m_w}}{\sqrt{\hat{v_w}} + \epsilon},$$

$$m_w^{(t+1)} \leftarrow \beta_1 m_w^{(t)} + (1 - \beta_1) \nabla_w L^{(t)},$$

$$v_w^{(t+1)} \leftarrow \beta_2 v_w^{(t)} + (1 + \beta_2) (\nabla_w L^{(t)})^2,$$

$$\hat{m}_w = \frac{m_w^{(t+1)}}{1 - \beta_1^{(t+1)}},$$

$$\hat{v}_w = \frac{v_w^{(t+1)}}{1 - \beta_2^{(t+1)}},$$

where β_1 and β_2 are the forgetting factors for gradients and second moments of gradients, respectively.

The difference between the Stochastic Gradient Descent algorithm and the Adam algorithms for learning is that Adam computes adaptive learning rates for each parameter. Adam stores an exponentially decaying average of past squared gradients v_t, similar to AdaGrad and RMSprop, in addition to an exponentially decaying average of past gradients m_t, that is analogous to momentum. All of these added parameters led Adam to train at a faster rate than Stochastic Gradient Descent.

Additionally, we defined our loss function to be a binary-cross-entropy loss function:

$$BCE = -(y \log(p) + (1 - y) \log(1 - p)),$$

where $M = 2$, y is the binary indicator (0 or 1) if the class label predicted for the observation is correct, and p is the predicted probability that the observation is of the labeled

class. We chose to use binary cross-entropy as our loss function because we treated our problem as 5 different binary classification problems. We had to determine whether or not Atelectasis was present in the chest radiograph, whether or not Edema was present in the chest radiograph, whether or not Cardiomegaly, Consolidation, and/or Pleural Effusion were present in the chest radiograph. Because of the nature of the $1s$ and $0s$ in the problem statement, it seemed intuitive to use the binary cross-entropy loss function.

4. Evaluating Performance

To evaluate the performance of our models we looked into Area Under Receiving Operating Characteristics and accuracy, precision and recall, defined as such:

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})},$$

$$\text{Precision} = \frac{\text{TP}}{\text{FP} + \text{TP}},$$

$$\text{Recall} = \frac{\text{TP}}{\text{FN} + \text{TP}},$$

where TP, FP, TN, NP are the true positive, false positive, true negative, false negative values, respectively.

We ultimately decided on using accuracy, precision and recall since it was easier to implement and we have a class imbalance so it will allow us to visualize the performance in more important metrics. While we mostly used accuracy to determine how well our model was performing, we made sure to look at precision and recall to determine its effectiveness on predicting the disease vs predicting its not there, which is significantly easier due to class imbalance.

5. Dataset and Features

The dataset that we are using is the CheXpert dataset, provided by the Stanford Machine Learning Group. This dataset consists of 224,316 chest radiographs labeled for the presence of 14 common chest radiograph observations. Each observation was either labeled with $+1, 0, u$, *(blank)*, which indicated the presence of the observation, the absence of the observation, the *uncertainty* of the presence of the observation, and no mention of the observation, respectively.

5.1. Preprocessing

A lot of the preprocessing had to do with making the loading time faster. At the start, the images would take a long time to load, so eventually the images' arrays were stored and loaded instead. Additionally, one concern that needed to be addressed was how to approach the uncertainty labels as well as the *(blank)* labels. The CheXpert paper presents a number of different approaches to address the u labels, such as ignoring the u labels, turning them all into 1s or 0s, transforming the problem into a 3-class multilabel classification problem. For the baseline, we chose to mask all of the u labels as 0s, but for our final model, we chose to utilize the same approach that was mentioned in the CheXpert paper, where we masked u labels in the Atelectasis and Edema classes as $1s$, and we masked the u labels in the Cardiomegaly, Consolidation, and Pleural Effusion classes as $0s$ as those were the approaches that performed best on the CheXpert model.

Lastly, another major issue that we had to address is that poor breakdown between training and validation data once the data was separated into their individual views. The anteroposterior training dataset contained upwards of 160,000 scans, whereas the lateral and posteroanterior training datasets each contained roughly 28,000 scans. The anteroposterior validation set only contained 169 scans, $> 0.01\%$ of its training dataset. The lateral validation set only contained 32 scans, $> 0.001\%$ of its training dataset. Lastly, the posteroanterior validation dataset only contained 33 scans, $> 0.01\%$ of its training dataset. Additionally, there was no testing dataset. To address these issues, we had to parse our training dataset into a new smaller training set and the remaining part of the training dataset became our validation set. Lastly, the original validation set became our test set.

6. Experiments/Results/Discussion

6.1. Frontal AP Results

Our largest dataset belonged to the Frontal AP view so we decided to initially train on this dataset and then use transfer learning to converge faster and achieve higher accuracy on the Frontal PA view and Lateral view. Using DenseNet121, binary cross entropy loss, and evaluating our performance using accuracy, precision and recall. We managed to achieve an accuracy of 83.4%. This evaluates to 83.1% for Atelectasis, 88.4% for Cardiomegaly, 92.9% for Consolidation, 74.7% for Edema, and 77.6% for Pleural Effusion. We see promising results for the category Consolidation and okay results for Atelectasis and Cardiomegaly. On the other hand, our results for Edema and Pleural Effusion weren't great, but were expected. The original CheXpert paper also found difficulty in correctly predicting our results. On the other hand, when we evaluated using precision, we found an average of 52%. Similarly the categories Atelectasis, Cardiomegaly, Consolidation, Edema and Pleural Effusion achieved 40.1%, 55.4%, 35%, 57.6% and 73.7%. Finally, through recall we achieved an average of 35.3% and 2%, 44.1%, 1.5%, 54% and 73%. These results leave much to be desired but I strongly believe that with more time we can fine-tune such that we can achieve a more balanced model that will net us better results.

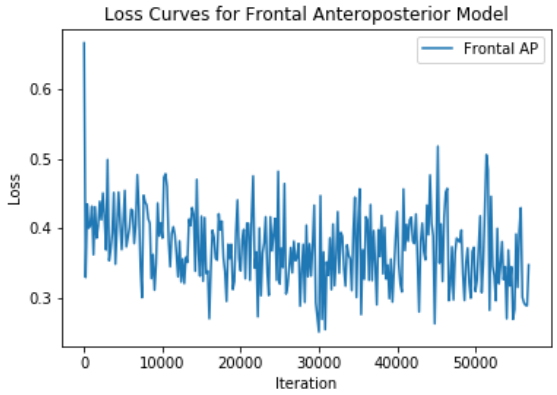

Figure 2. Loss curve for frontal AP model after 15 epochs

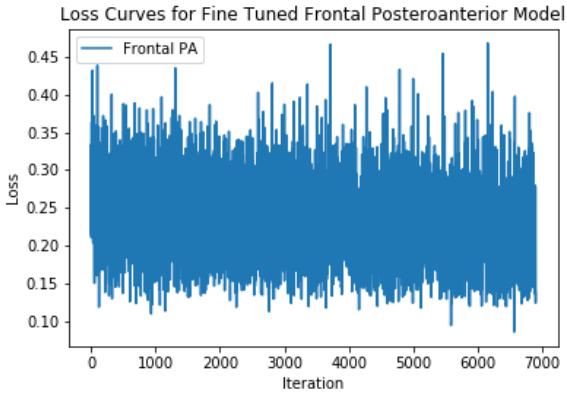

Figure 3. Loss curve for finely tuned frontal PA model after 10 epochs

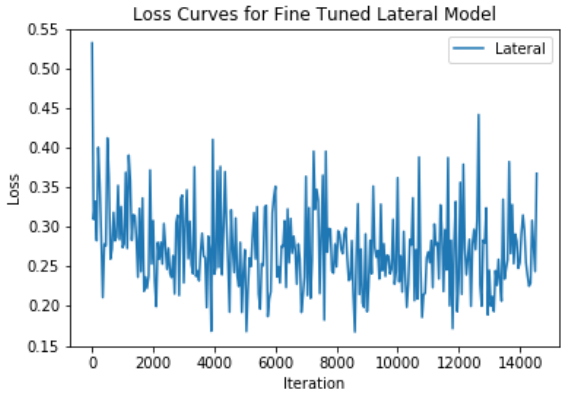

Figure 4. Loss curve for finely tuned lateral model after 10 epochs

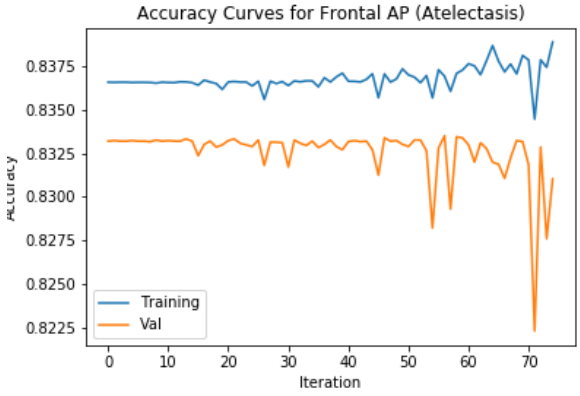

Figure 5. Accuracy curve for frontal AP model (atelectasis)

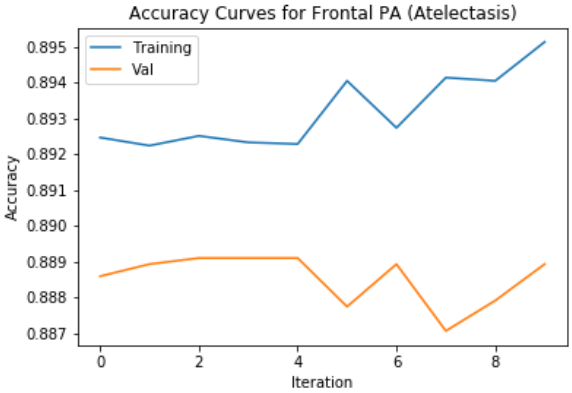

Figure 6. Accuracy curve for frontal PA model (atelectasis)

6.2. Transfer Learning

While we acknowledge that our base model trained with Frontal AP results left much to be desired, we still decided to see if transfer learning could help us train models for Frontal PA and Lateral views. We found similar results that need significant improvements, and we believe that our next step moving forward should be trying to achieve a better Frontal PA view model.

6.3. Results

Below we will include our loss curves and some accuracy curves we saw during our training process. We noticed that that our loss curve is slowly going down but also quite noisy. Similarly our accuracies are going up slightly, but I would account for that in terms of noise instead of any significant improvements found by our models.

7. Conclusion/Future Work

Our problem at task was to determine whether or not view-specific models would perform better than a model that was trained on all views. Overall, we saw some improved performance on some of the observations and decreased performance on other observations. Based off of these results, we cannot conclude that utilizing view-specific models is a better approach than just designing a model trained on all views. However, the use of transfer learning does permit the view-specific models for the lat-

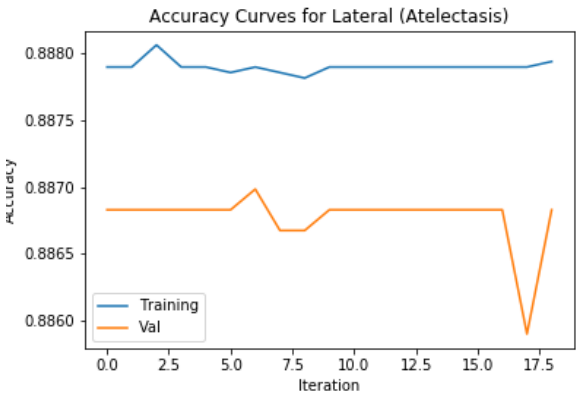

Figure 7. Accuracy curve for lateral model (atelectasis)

eral and frontal posteroanterior more room to grow as researchers can develop a more complex system.

Additionally, for this project, we implemented two architectures, AlexNet and DenseNet121, but finding the best architecture for this dataset is still an open research problem presented by CheXpert [5]. For future work, we still have to develop visualizations for the AUROC values in addition to the precision and recall metrics. Additionally, we hope to experiment with additional, more complex architectures such as ResNeXt and ResNet.

Finally, our biggest issue seems to arise from class imbalance which doesn't allow the model to correctly and easily learn. Some techniques we could utilize to attempt to resolve this issue are data augmentation and class weights.

8. Acknowledgments and Contributions

We would like to acknowledge Pranav Rajpurkar and the Stanford Machine Learning Group for introducing us to this project and for providing us access to this dataset.

Stephen and Miguel contributed equally in the data preprocessing tasks involved. Afterwards, Stephen contributed a lot on the implementation of the DenseNet121 model for the frontal anteroposterior model whereas Miguel focused on setting up the transfer learning for the secondary models and adjusting the model's parameters such as the optimization algorithms, the learning rates, the weight decays, and more to better suit our needs. Stephen and Miguel were both in constant communication throughout the entire project to ensure that both of us knew exactly what was going at all times.

References

[1] In radiology, turnaround time is king. https://www.diagnosticimaging.com/pacs-and-informatics/radiology-turnaround-time-king. Accessed: 2019-05-14.

[2] Repo for replication of chexpert paper and submission to chexpert competition. https://github.com/simongrest/chexpert-entries. Accessed: 2019-05-15.

[3] K. Doi. Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. *Computerized Medical Imaging and Graphics*, 31(4):198 – 211, 2007. Computer-aided Diagnosis (CAD) and Image-guided Decision Support.

[4] G. Huang, Z. Liu, and K. Q. Weinberger. Densely connected convolutional networks. *CoRR*, abs/1608.06993, 2016.

[5] J. Irvin, P. Rajpurkar, M. Ko, Y. Yu, S. Ciurea-Ilcus, C. Chute, H. Marklund, B. Haghgoo, R. L. Ball, K. Shpanskaya, J. Seekins, D. A. Mong, S. S. Halabi, J. K. Sandberg, R. Jones, D. B. Larson, C. P. Langlotz, B. N. Patel, M. P. Lungren, and A. Y. Ng. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. *CoRR*, abs/1901.07031, 2019.

[6] N. M. Khan, N. Abraham, and L. Guan. Machine learning on biomedical images: Interactive learning, transfer learning, class imbalance, and beyond. *CoRR*, abs/1902.05908, 2019.

[7] S. Kornblith, J. Shlens, and Q. V. Le. Do better imagenet models transfer better? *CoRR*, abs/1805.08974, 2018.

[8] M. Raghu, C. Zhang, J. M. Kleinberg, and S. Bengio. Transfusion: Understanding transfer learning with applications to medical imaging. *CoRR*, abs/1902.07208, 2019.

[9] M. M. Rahman and D. N. Davis. Addressing the class imbalance problem in medical datasets. 2013.

[10] B. van Ginneken. Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. *Radiological Physics and Technology*, 10(1):23–32, Mar 2017.