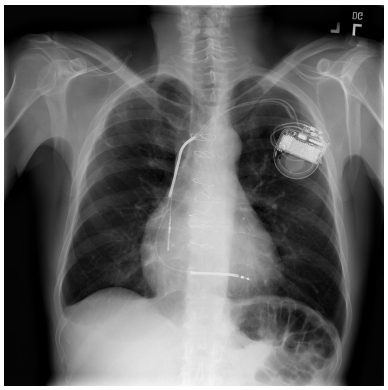
Final Report

By: Amulya Cherian, Hilary Present, and Nabil Rmiche

Introduction

Radiologists are responsible for reading and interpreting hundreds of images per day. This intense and tedious workload can easily result in human error. Inaccurate diagnostic radiology can lead to “treatment delays, poor outcomes, higher healthcare costs” for the patient (GE Healthcare, 2023). A study by the American Medical Association (AMA) found that 40.2% of the radiologists out of the 3,500 physicians in their sample have been sued in their career so far (Guardado, 2023). In addition, “errors in diagnosis” are the most common cause of “malpractice suits against radiologists,” further pointing to the need for a transformative improvement in diagnostic radiology (Whang et al., 2013).

Incorporating machine learning and AI into radiology can help alleviate some of the pressure that radiologists experience. For instance, automating image processing and extraction of quantitative information from medical imagery can help streamline the process of interpreting imagery for radiologists. For our project, we built a Convolutional Neural Network (CNN) model to predict thoracic diagnosis for chest X-ray images such as the image (# 00000013_044) below. This image corresponds to the following diagnoses: cardiomegaly, mass, and pleural thickening.



Related Works

Deep Learning for Medical Images:

[1] E. Çallı, E. Sogancioglu, B. van Ginneken, K. G. van Leeuwen, and K. Murphy, "Deep learning for chest X-ray analysis: A survey," *Medical Image Analysis*, vol. 72, p. 102125, Aug. 2021, doi: https://doi.org/10.1016/j.media.2021.102125.

- This paper applies image-level prediction with both classification and regression models, segmentation, image generation, and localization for chest X-ray datasets. This article is especially helpful for our project because it gives us guidance on what to consider when training models on chest X-ray data, especially when medical imagery varies so much across different tissue matters and different patients.

[2] X. Chen *et al.*, "Recent advances and clinical applications of deep learning in medical image analysis," *Medical Image Analysis*, vol. 79, p. 102444, Jul. 2022, doi: https://doi.org/10.1016/j.media.2022.102444.

- This paper is relevant to our project since it involves the challenge of limited large-sized, well-annotated datasets in medical image analysis, which is an obstacle we were facing when we first began looking for datasets to begin our project. By reviewing recent advancements in unsupervised and semi-supervised deep learning for medical imaging, this paper provides insights that help to inform our approach for improving our models' performance, even with some dataset constraints.

[3] S. T. H. Kieu, A. Bade, M. H. A. Hijazi, and H. Kolivand, "A Survey of Deep Learning for Lung Disease Detection on Medical Images: State-of-the-Art, Taxonomy,

Issues and Future Directions," *Journal of Imaging*, vol. 6, no. 12, p. 131, Dec. 2020, doi: https://doi.org/10.3390/jimaging6120131.

- This paper by Kieu et al. is relevant to our project because it offers an overview of deep learning methods for lung disease detection in medical images. Insights from this survey can guide our data augmentation, algorithm selection, and transfer learning approaches, which will ultimately assist in enhancing our deep learning models for pulmonary disease diagnosis.

TorchIO:

[4] F. Pérez-García, R. Sparks, and S. Ourselin, "TorchIO: A Python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106236, Sep. 2021, doi: https://doi.org/10.1016/j.cmpb.2021.106236.

- This paper by Pérez-García et al. on TorchIO is relevant to our project since it provides a comprehensive overview to efficiently handle medical images, which is necessary for our project with the NIH chest X-ray dataset. Borrowing the techniques seen in this paper – TorchIO's capabilities for preprocessing, augmentation, and patch-based sampling – will allow us to streamline our data processing pipeline and enhance the performance of our deep learning models.

YOLO Algorithm:

[5] N. Palanivel, Deivanai S, Lakshmi Priya G, Sindhuja B, and Shamrin Millet M, "The Art of YOLOv8 Algorithm in Cancer Diagnosis using Medical Imaging," Nov. 2023, doi: https://doi.org/10.1109/icscan58655.2023.10395046

- This study implements the You Only Look Once (YOLO) v8 method for early diagnosis of different types of cancer such as leukemia, skin cancer, cervical cancer, and lung cancer. This article was selected because the dataset is quite diverse and consists of different types of imaging. For instance, the dataset includes Pap smear images of cervical cancer, blood smear images of leukemia, and histopathological images of lung cancer. YOLO v8 works by dividing up each image into a matrix of cells and then predicting the class of each detected object, which should apply well for our project with the chest X-ray dataset.

[6] A. Agarwal, P. Sharma, P. Awasthi, and D. Arora, "Tumor Segmentation and Detection using Convolutional Neural Network," 6th International Conference on Computing for Sustainable Global Development (INDIACom), pp. 471–474, Feb. 2020, Available: https://ieeexplore.ieee.org/abstract/document/8991316

- This paper by Agarwal et al. is helpful for our project as it examines how to detect tumors with a CNN model using Pytorch libraries such as torch.nn, torch.autograd, and torch.vision. Although this paper focuses on MRI imagery, the techniques utilized in building a CNN model can translate to the X-ray images that we are using in our project.

[7] Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., & Summers, R. M. (2017). Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2097-2106). doi: https://doi.org/10.48550/arXiv.1705.02315

- This paper introduces the NIH dataset used in our project. It details precisely the collection process and as such gives us insights regarding how to make the best use of it. It also provides us with the quantitative results we chose to use as our baseline.

Methods

Initially, we processed our dataset by resizing the images to a uniform size and converting them to PyTorch tensors, ready for model input. As part of our preprocessing, we also implemented a series of transforms to the images, which included resizing them, random rotations of up to 15 degrees, random resized cropping, and horizontal flipping to augment the data variability. These transforms help incredibly when it comes to training robust models that are able to handle the real-world variations in X-ray images.

We began by splitting the dataset into training and validation sets using an 80-20 ratio, though for future improvements, adjusting to a 70-15-15 split for training, validation, and testing datasets respectively could enhance our model comparison and accuracy validation.

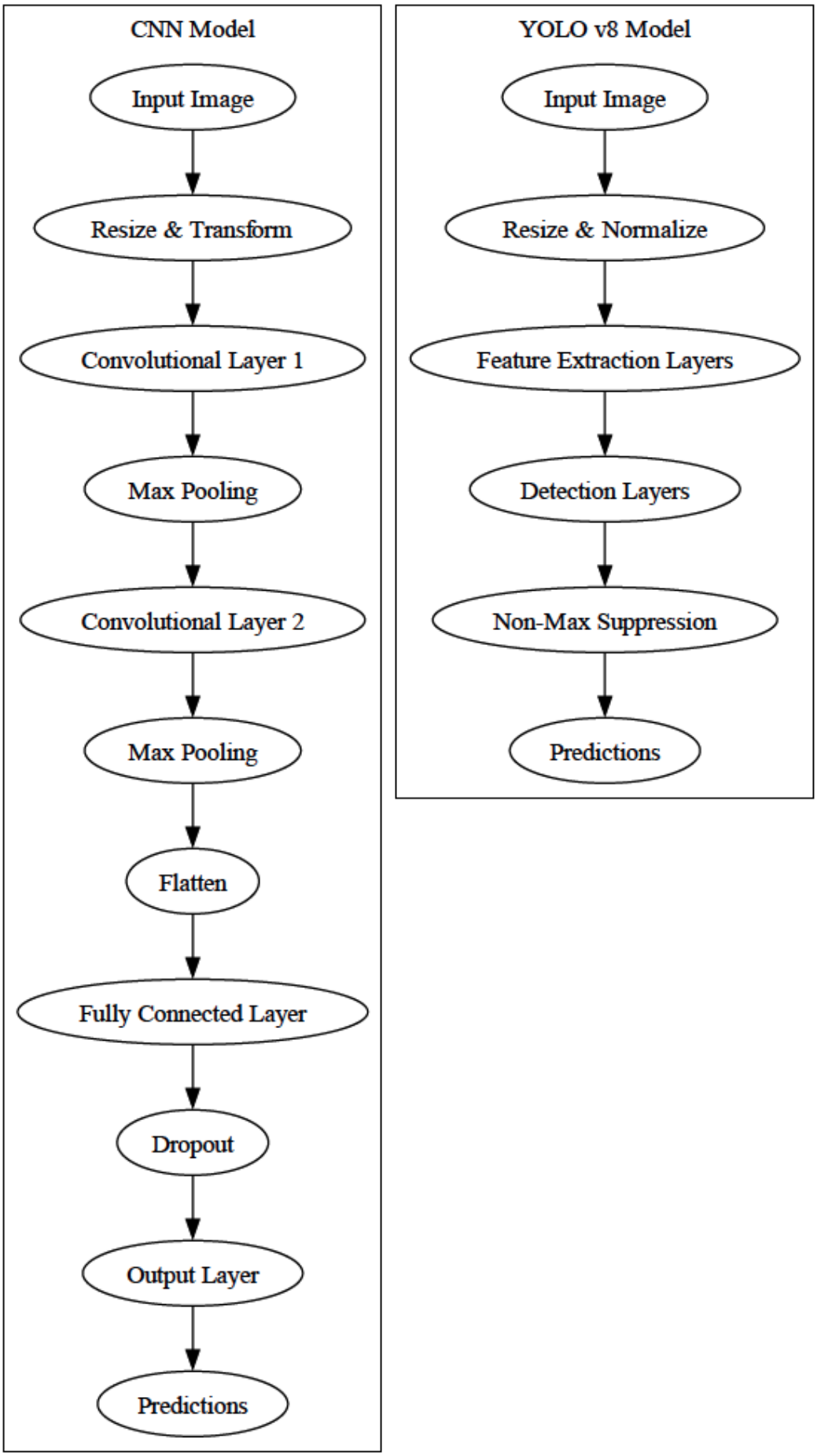
In addition to the initial data preparation, we conducted exploratory data analysis (EDA) to check the quality of our data and to better understand the distribution of labels. The original labels have been extracted from radiology reports thanks to an NLP tool with a reported accuracy of 90%. Because of this, a significant proportion of the dataset is multi-labelled and since the original reports were never publicly released, we chose to treat them all as equally likely. During the EDA process, we first checked for null values within the ‘Finding Labels’ column. Then, created a new column ‘Labels’ to handle instances where images have multiple labels – which were strings concatenated with '|' in the ‘Finding Label’s column. The ‘Labels’ column split the

concatenated labels into lists of labels, allowing for better processing. Finally, we tallied the counts for each label to ascertain which categories had the most and the least data, which is crucial for understanding the balance of our dataset and informing potential strategies for model training and evaluation. To finalize our EDA, we constructed a pivot table to visualize the distribution of patients across each gender, segmented into age groups spanning 10 years each.

For the first model, we implemented a simple convolutional neural network (CNN) with two convolutional layers for feature extraction. These were followed by a subsampling max-pooling layer to reduce spatial dimensions, a dropout layer for regularization to prevent overfitting, and fully connected layers to combine features from the previous layers to make the final label predictions. We trained this CNN for 10 epochs using a cross-entropy loss function and Adam optimizer, with a learning rate of 0.001.

Our second method that we implemented involved deploying all our pipeline onto the PACE-ICE cluster. We were able to train Yolov8 for 50 epochs with mostly default parameters.

In both methods, we handled the labeling uncertainty in two different ways. We either directly predicted multiple labels per sample or we performed a standard multi-class classification by untangling the dataset, that is repeating samples with more than one label and assigning a unique label to each one. The diagram below shows a detailed overview of the pipeline for each of our models:

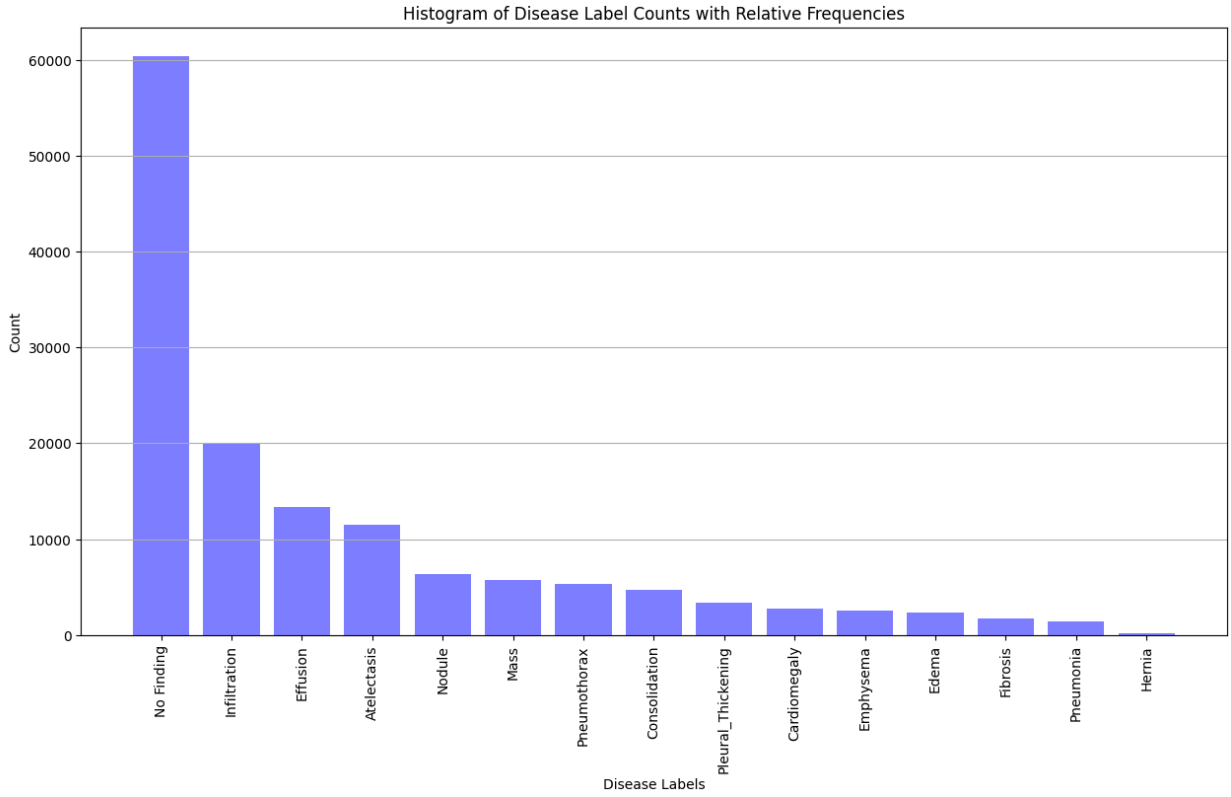

CNN Model
Input Image
Resize & Transform
Convolutional Layer 1
Max Pooling
Convolutional Layer 2
Max Pooling
Flatten
Fully Connected Layer
Dropout
Output Layer
Predictions
YOLO v8 Model
Input Image
Resize & Normalize
Feature Extraction Layers
Detection Layers
Non-Max Suppression
Predictions

Experiment Setup

We are utilizing the Clinical Center Chest X-Ray dataset from the National Institute of Health (NIH). The dataset consists of 112,120 frontal-view chest X-ray PNG images along with the metadata for these images such as the image index, diagnostic finding labels, view position, and original dimensions. All of the X-ray images have 1024*1024 resolution. The dataset also has patient information that corresponds to each image such as patient ID, age, gender, and number of followup appointments. These X-Ray images come from 30,805 unique patients, which makes for a large sample size. The table below highlights the breakdown of patient demographics within the NIH dataset.

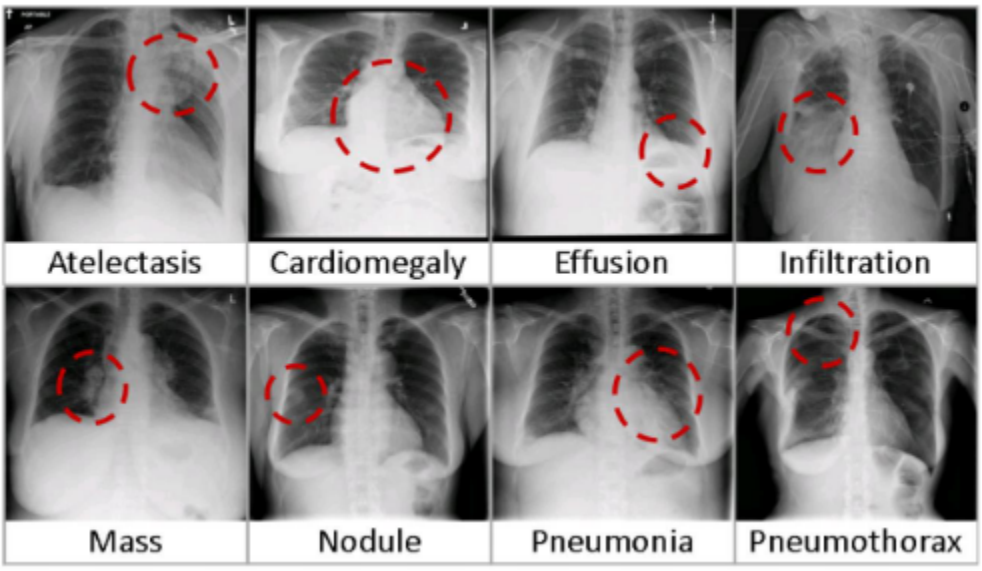
Age Group	# of Female Patients	# of Male Patients
0-10	616	922
10-20	2350	3158
20-30	5258	7597
30-40	7776	8762
40-50	10387	11515
50-60	11843	15498
60-70	7566	11399
70-80	2541	3913
80-90	410	545
90-100	33	31

There are a total of 15 disease labels, including a ‘No Finding’ one that may or may not be a healthy sample, but each image can be associated with more than one disease label. There are 836 unique labels as a result of X-ray images with multiple diagnoses. This dataset was chosen because it is fairly representative of the real patient population distributions and has a



variety of thoracic pathologies as shown in the histogram above.

The figure below distinguishes between some of the different thoracic pathologies found on chest X-rays (Wang et al., 2017).



The aim of our project is to build a model that processes X-ray images and predicts one or more of the fourteen disease labels as the corresponding thoracic diagnosis. To measure the quality and performance of our model, we examined the following metrics: accuracy and cross-entropy loss. In our efforts to optimize for maximizing accuracy and minimizing loss, we primarily focused on two models – a convolutional neural net (CNN) and a Yolo v8 algorithm. During the initial training of our CNN model, we monitored the loss and accuracy on both the training and validation datasets. Over time, both the training and validation losses showed a downward trend, suggesting effective learning. Ultimately, the training accuracy consistently ranged between 0.93 and 0.95, while the final validation accuracy stood at 0.4204.

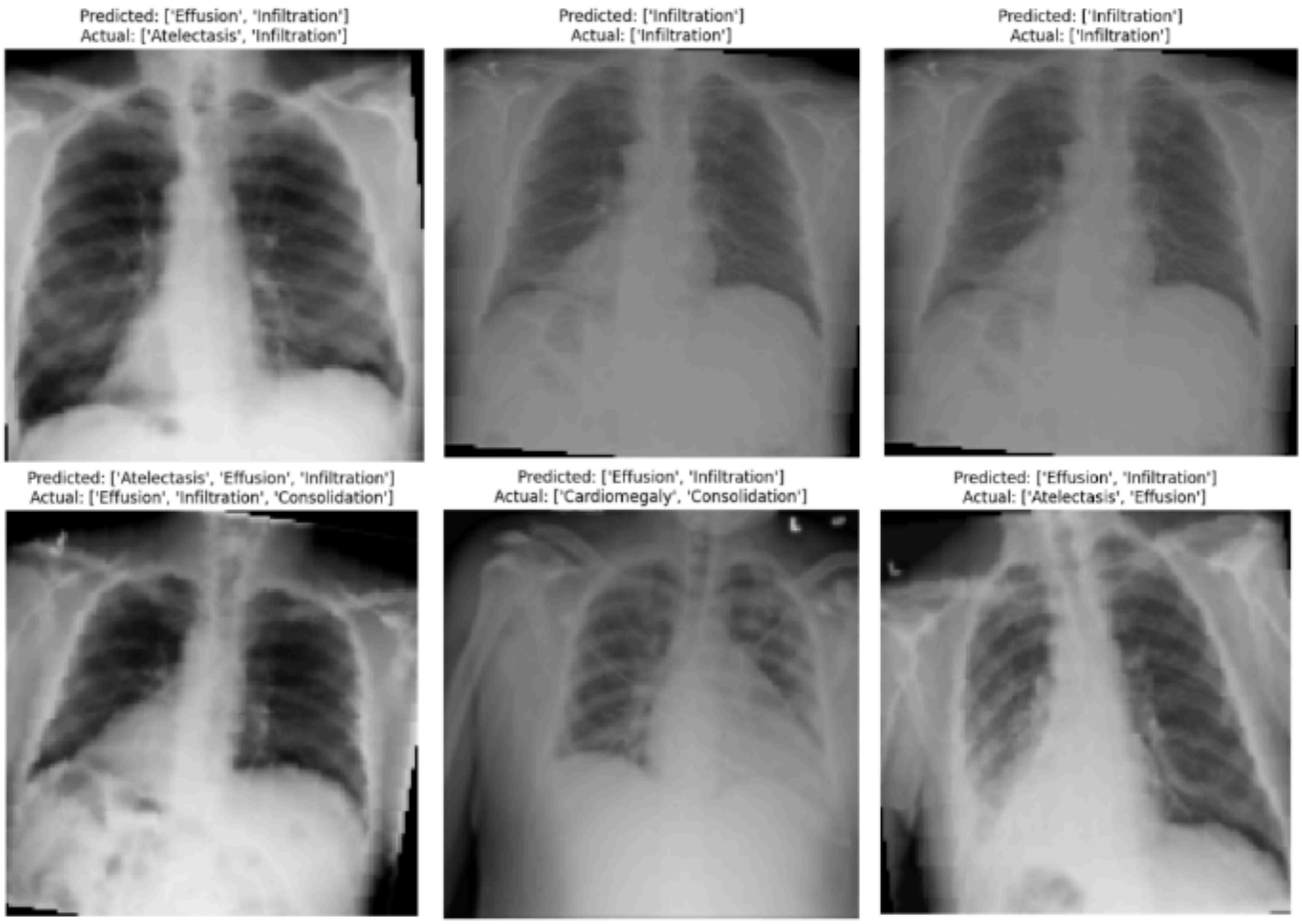
Further experimentation was conducted using the Yolo v8 algorithm on our cluster. The main objective of this test was to ensure our pipeline functioned properly end-to-end and to assess the costs – both in computation and in time – of training a sophisticated model on a substantial dataset.

Our review of existing literature revealed a focus primarily on the image data for algorithmic diagnosis. To address this gap, we incorporated additional patient information, such as age and gender, alongside the image data. This approach involved developing the CNN model to utilize both the X-ray image data and patient demographic data from a CSV file to enhance the accuracy of our diagnostic predictions. Incorporating patient demographic data such as age and gender alongside the image data allows us to build a more robust model. This makes sense, because that data proves additional context that can be crucial for accurate diagnostics. For example, certain thoracic diseases may present differently across age groups or genders, and this multi-dimensional analysis can lead to more precise diagnostic results, filling an important gap in the current research that exists today. We also considered using Yolo v8 to potentially get better

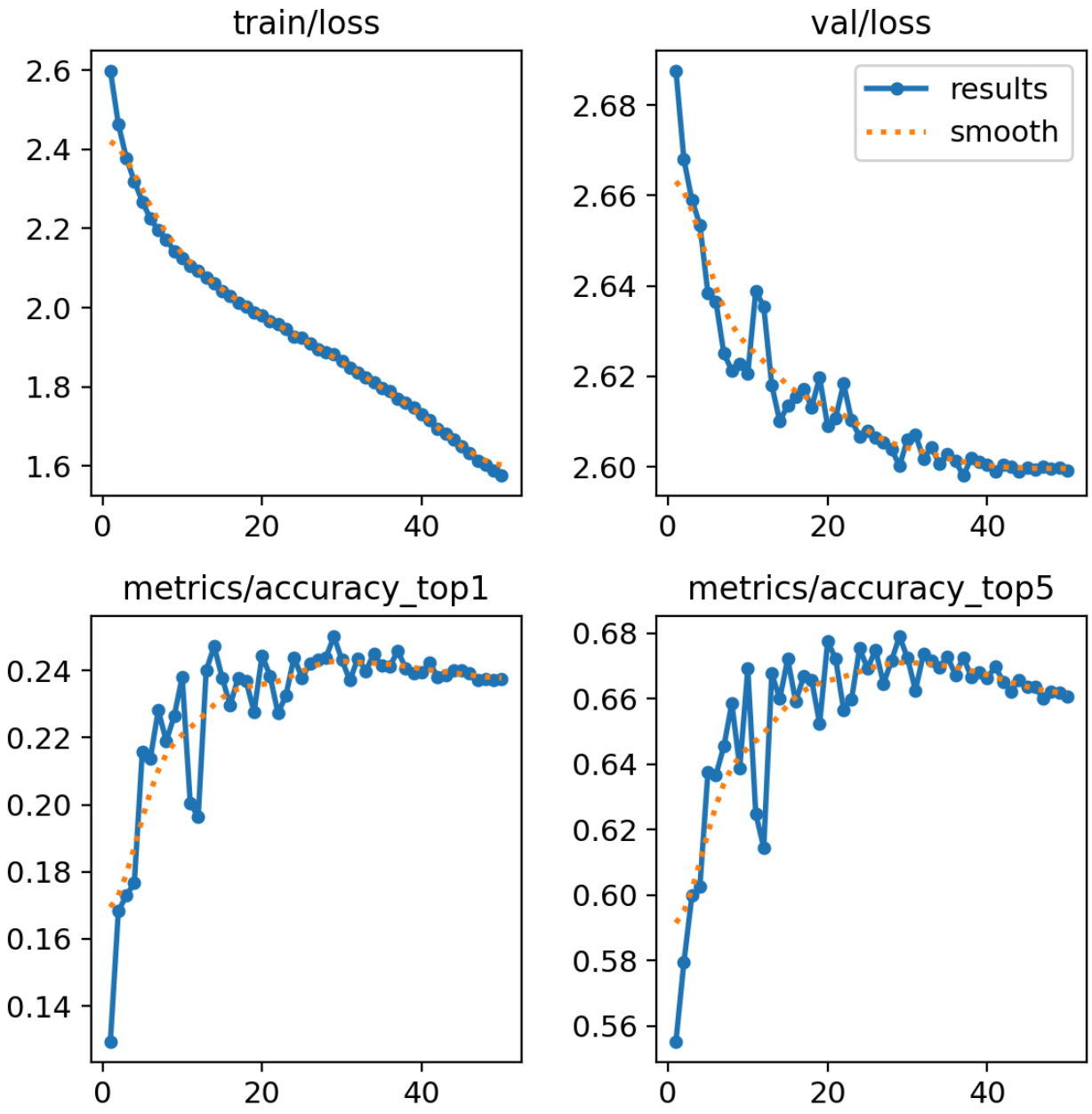
features representation from image data thanks to its state of the art Darknet backbone as well as its advanced augmentation techniques like Mosaic or Mixup.

Results – CNN Model

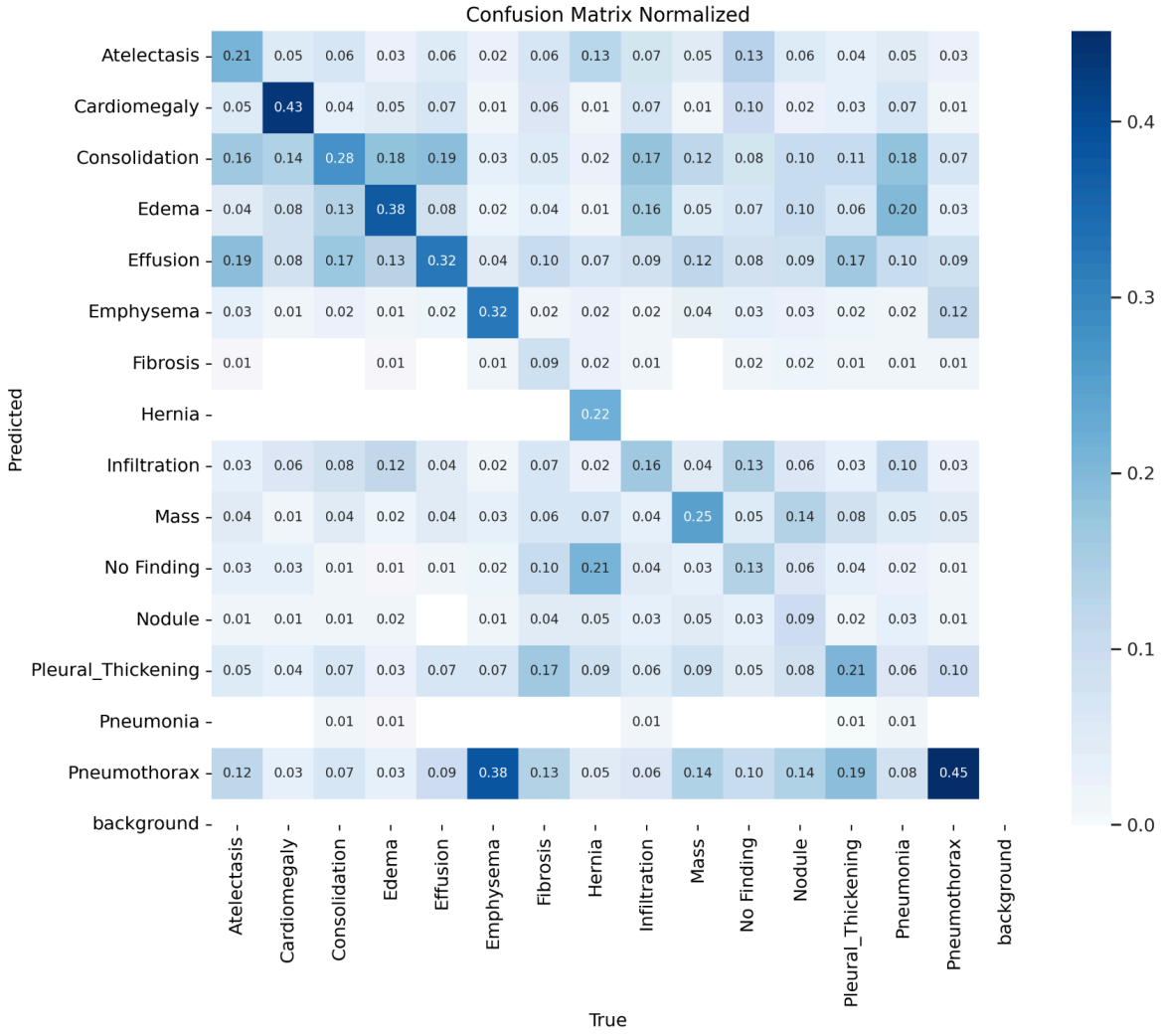
Epoch #	Training Loss	Training Accuracy	Validation Loss	Validation Accuracy
1	0.1831	0.0296	0.1752	0.0297
2	0.1807	0.0296	0.1743	0.0297
3	0.1789	0.0296	0.1728	0.0297
4	0.1778	0.0296	0.1722	0.0297
5	0.1768	0.0296	0.1718	0.0297



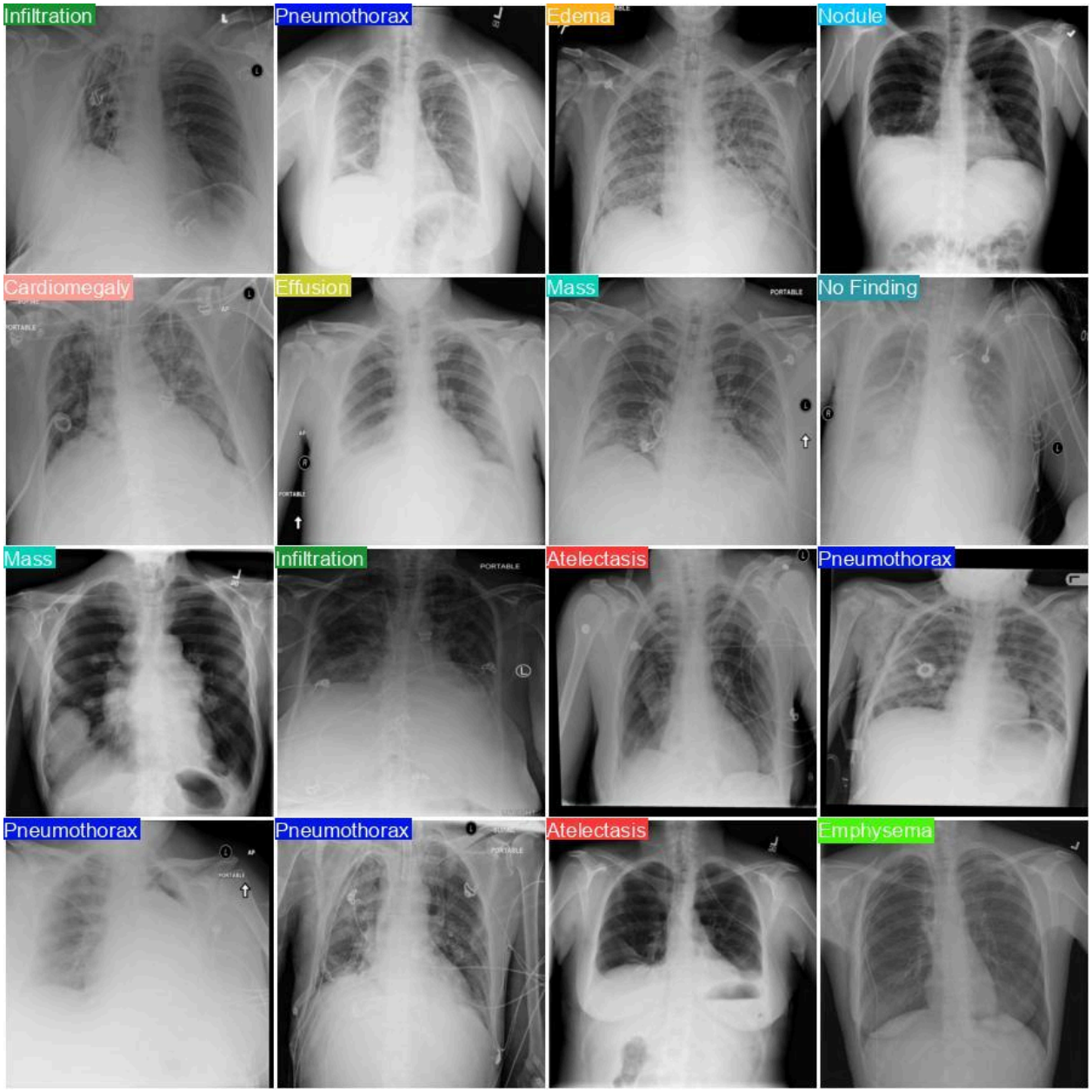
Results – Yolo v8 Model



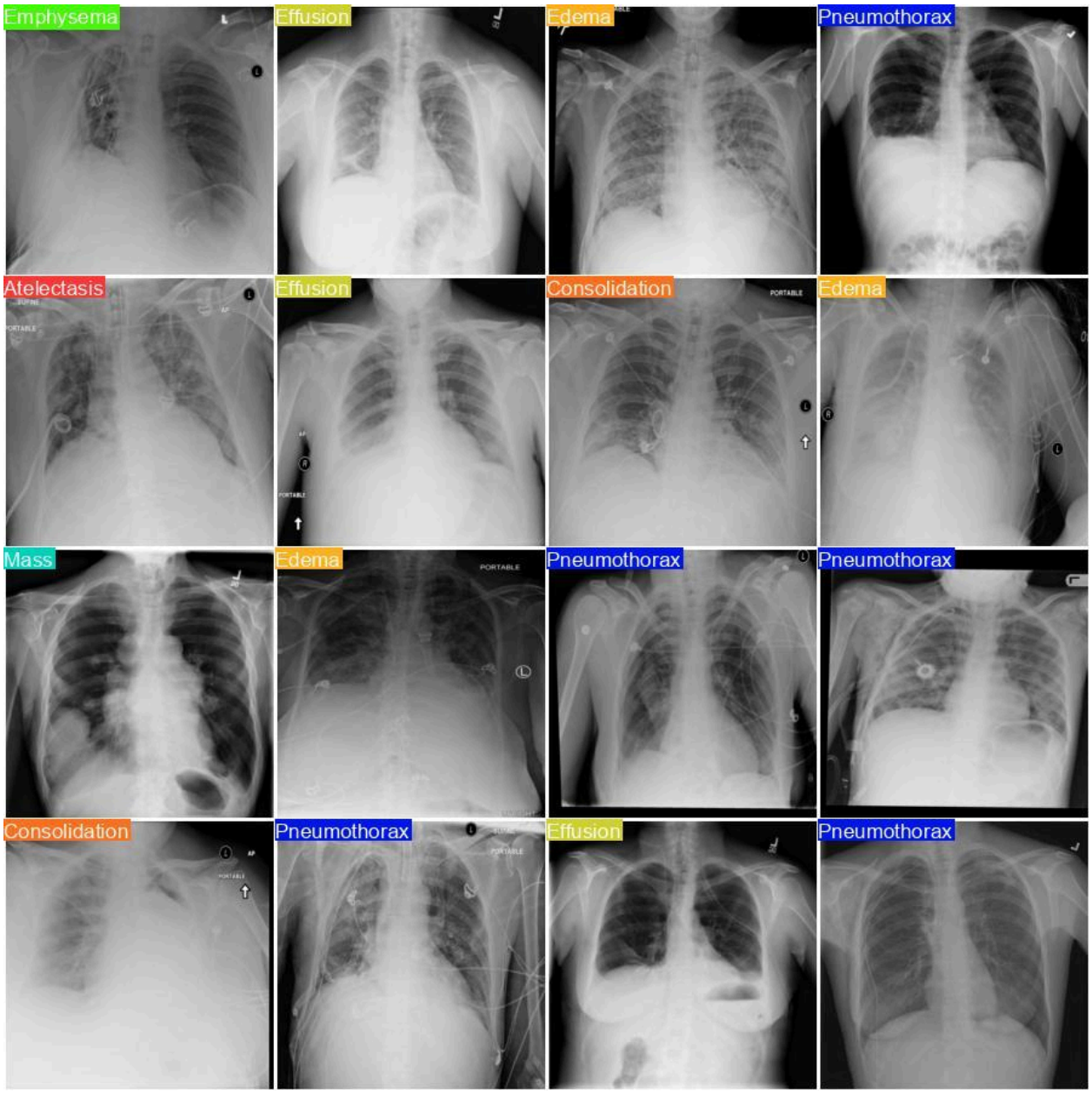
Yolov8 training



Yolov8 classification performance



Ground truth labels



Predicted labels

Discussion

As a baseline, we used the results from the original NIH dataset paper [6], which reported an accuracy of 70% across the 8 initial diseases. Despite our efforts, our models did not achieve the expected results; our best results came from our CNN model, with a validation accuracy of

0.42 compared to a final validation accuracy of 0.24 for our YOLO v8 algorithm. Ultimately, we were able to identify several key factors that explain these outcomes:

- To train our models within a feasible time frame and using a manageable amount of computing resources for the scope of this project, we had to use smaller image sizes – 128x128 for the CNN model and 224x224 for Yolov8. This is in contrast to the baseline approach we were comparing our models to, which used full-resolution images (1024x1024). Our qualitative results showed that this dataset is very challenging even for experts. The issues with noisy labeling are not only from the NLP tool, but also from radiologists using different terms to refer to the same conditions based on their own clinical contexts and histories. While we did try to mitigate the impact of this by adding transformations to our input images (random flipping, random rotations, etc.), we saw how difficult it is to expect robust performances from models trained on low-resolution images, particularly when the area of interest in a typical chest x-ray is relatively small.
- Initially, we chose not to address the data imbalance with custom loss functions, such as focal loss, although such methods were employed in the baseline approach, including weighted cross-entropy. Our primary goal was to compare different architectures – CNN, ViT, and a diffusion classifier. Therefore, it made more sense to address and rebalance our dataset rather than modifying the complex codebase of each model to implement a custom loss. However, despite achieving decent top-5 accuracy with this smaller, balanced dataset, we still found factors such as our smaller image size and the inconsistent labeling impeded our ability to achieve comparable predicted label accuracy to the baseline NIH paper [6].

Challenges Encountered

Throughout this project, we faced numerous challenges that hindered our ability to quickly prototype and implement potential solutions. Some team members encountered hardware limitations, as PACE-ICE requires a VPN connection over eduroam, which proved to be unreliable at times. Additionally, when we used the cluster each training epoch took roughly 5 times longer to run when executed via a batched Slurm job compared to an interactive session. It took some time for us to realize that this discrepancy was the root cause of many failed jobs.

Moving Forward

To enhance our methods and outcomes in future iterations of this project, several steps could be taken:

1. We could increase the reliability of our training data through more rigorous validation of labels, potentially augmenting our dataset with other, more annotated image databases to create a more robust train set.
2. We could use techniques like SMOTE to create synthetic data, helping to balance class distributions and enhance our model robustness.
3. We could do a 70-15-15 split as opposed to a 80-20 split for training, validation, and testing. This would allow for more thorough model evaluation and refinement, improving our ability to compare our models and to better determine the final model's accuracy.
4. To overcome the hardware and connectivity issues we experienced, we could either dedicate more time to allow our models to train, or could explore more robust infrastructure or alternatives to PACE-ICE, which potentially could improve efficiency.

5. Additionally, we could address the inefficiencies we saw in batched Slurm jobs, optimizing our code for batch processing or communicating with the IT department to better understand and mitigate these issues.

If we were to start this project from scratch, we would begin with a thorough initial analysis of the dataset and identify potential computational bottlenecks, which could preemptively address many of the challenges we faced. Additionally, we could spend more time searching for relevant annotated datasets that we could augment with the NIH dataset we focused on, hopefully making our final model more robust. This would enhance the diversity and accuracy of our training data, providing a solid foundation for model training. Furthermore, if we were to start this project over, we would undergo more rigorous testing on each iteration of our models. This would involve spending more time fine-tuning our initial data transformations (beyond random flips and rotations), as well as tuning the hyperparameters to enhance each model's performance. This would allow us to ensure their accuracy and efficiency are maximized before selecting a final model.

Github: https://github.gatech.edu/hpresent6/6476_Project

References

GE Healthcare. (2023, June 13). Improving Accuracy in Radiology Images and Reports. GE Healthcare. https://www.gehealthcare.com/insights/article/improving-accuracy-in-radiology-images-and-reports

Guardado, J. R. (2023). Medical Liability Claim Frequency Among U.S. Physicians. American Medical Association: Policy Research Perspectives.

Whang, J. S., Baker, S. R., Patel, R., Luk, L., Castro, A. (2013, February 1). The Causes of Medical Malpractice Suits against Radiologists in the United States. Radiology, 266(2). https://doi.org/https://doi.org/10.1148/radiol.12111119

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, Ronald M. Summers. ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly- Supervised Classification and Localization of Common Thorax Diseases, IEEE CVPR, pp. 3462-3471,2017