Lung Disease Diagnosis

Names: Brian Ellis, Jamys Solosky
PSU IDs: 901145253, 918847107
Emails: bje5256@psu.edu, jjs7331@psu.edu

1. INTRODUCTION

Our project focuses on lung prediction, specifically using Convolutional Neural Networks (CNNs) to analyze medical x-rays of lungs. The goal is to classify these x-rays into one of three categories: normal, virus, or bacteria. We anticipate several challenges along the way.

First, we have the advantage of pre-sorted data, which simplifies our preprocessing tasks. However, selecting the right model, especially for transfer learning, is crucial. The choice of our model base will impact both the accuracy of our predictions and the processing speed. Therefore, initial research on available models for transfer learning is essential.

Another potential challenge is the robustness of our dataset. If it falls short in providing a solid foundation for our model, we may need to perform data preprocessing, such as image alterations and generating additional training samples. This will help ensure that our model becomes more robust and capable of making accurate predictions.

Furthermore, we might encounter issues with our model's ability to generalize effectively across all classes. For instance, our model may excel at detecting normal lungs but struggle with virus classification. In such cases, we must adjust our training focus, possibly giving more emphasis to the virus category to achieve better balance.

Lastly, there is the possibility that our model might not produce remarkable results, making it challenging to justify documentation or presentation. If our model's performance falls short of expectations, and its image classification isn't sufficiently accurate, we might need to reevaluate our approach and experiment with different techniques.

2. RELATED WORK

In 2018, Timor Kadir and Fergus Gleeson published a paper that delved into the application of computer vision and machine learning techniques for diagnosing lung cancer in medical patients. In a prior study conducted in 2015, the authors had employed a Support Vector Machine (SVM) for the task of classifying lung cancer in X-ray images. Unfortunately, the results they obtained at that time were only marginally better than random chance. It's important to note that this was before the widespread adoption of Convolutional Neural Networks (CNNs) in computer vision projects.

The authors also discuss the disparities in feature selection and extraction methods between CNNs and traditional machine learning approaches (1).

In researching related work, we thought it would be beneficial to understand the diagnostic accuracy of medical professionals without AI assistance. A paper published in 2020 by L. Arts et al., aimed to evaluate the diagnostic accuracy of lung auscultation, (listening for abnormalities in patient breathing), for common respiratory pathologies in adults with respiratory symptoms. Among 34 studies included, the overall pooled sensitivity for lung auscultation was found to be 37%, with a specificity of 89% (2).

Despite the potential utility of lung auscultation, the findings revealed a low sensitivity across various clinical settings and patient populations, limiting its diagnostic efficacy. The analysis also indicated significant heterogeneity and suggested the presence of publication bias in the studies considered. The study further highlighted associations between sensitivity and specificity with various factors, including the diagnosis group, index test used, department, percentage male, and average age of the study sample. These results emphasize the challenges and limitations of relying solely on lung auscultation for respiratory pathology diagnosis, particularly in comparison to alternative diagnostic modalities (2).

3. PROPOSED METHOD

Based on some initial research, we narrowed our focus to 4 potential deep learning Convolutional Neural Networks to use as transfer learning models:

- Resnet
- VGG-16
- Inception (GoogLeNet)
- EfficientNet

After doing some initial tests with each model, we decided to focus on GoogLeNet and its modular extension InceptionV3.

GoogLeNet is a type of convolutional neural network based on the Inception architecture that is 22 layers deep. It utilises Inception modules, which allow the network to choose between multiple convolutional filter sizes in each block. An Inception network stacks these modules on top of each other, with occasional max-pooling layers with stride 2 to halve the resolution of the grid.

Inception v3 is an image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset. The model itself is made up of symmetric and asymmetric building blocks, including convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. Batch normalization is used extensively throughout the model and applied to activation inputs. Loss is computed using SoftMax.

4. EXPERIMENTS

a. Dataset

We used the lung prediction dataset provided by the instructor. The dataset is composed of medical x-rays with 3 classes for lungs: normal, virus, and bacteria. The dataset is composed of over 1,700 images and is equally balanced among all 3 classes. Because the dataset is so large, we uploaded the data to our google drives and mounted our drives in the google colab environment.

b. Baselines

Our baseline SVM model was trained on extracted features from our InceptionV3 transfer learning model. The SVM classifier performed no better than random chance.

c. Experimental Results Analysis

After mounting our google drives, we build directories to store train and validation files of each class. After defining a 70/30 train test split, we move the correct proportion of each class into their respective train and validation folders. At the end of this process, around 390 images of each class are in their training folders, and ~ 166 images are in their validation folders.

After splitting our dataset, we use PyTorch to build our model using GoogLeNet. We apply several data augmentations to the images including random vertical and horizontal flips, image rotations, and scale them to fit the ImageNet format 224x224 since that is the dataset our transfer learning model is built upon. After defining the training procedure, we build our base model using GoogLeNet architecture and replace the final layer with a 6 unit layer and include a 3 unit output layer using SoftMax activation. We define our criterion using Cross Entropy Loss and our optimizer as Stochastic Gradient Descent with a learning rate of 10^-3. We additionally add a decay learning rate every 7 epochs.

Once our PyTorch model was finally defined, we run the model for 7 epochs and achieve a best validation accuracy of 72%, performing exceptionally well on the bacteria and normal classes while underperforming in the virus class. We notice that this 7 epoch training took 74 minutes, which hampers us with computational limitations if we want to train into several more epochs.

After multiple attempts to increase the performance of our model using PyTorch, we move to TensorFlow to see if we can get any different results. We import the InceptionV3, a module derived from GoogLeNet, and set the weights as those of ImageNet, similar to the PyTorch implementation. We add a final layer with 6 neurons and top it off with a 3 unit SoftMax layer. We use Adam optimizer instead of SGD and we also decrease the resolution of the images to 128x128 instead of 224x224 to speed up training. We found minimal performance decrease with this adjustment and were able to perform more epochs.

We achieve a remarkable 77.5% accuracy over 7 epochs with this new model even with our input sizes reduced. We can notice that the model has not converged yet, so it might be beneficial to implement more epochs. Because of this initial improvement, we attempt to increase the epochs to see when the model will converge. Unfortunately, we do not gain much better performance by increasing the epochs. The model seems to converge around this upper 70s validation accuracy range. We attempt to increase the model complexity and achieve a best validation accuracy of 79%, but after several more epochs the

model begins to overfit and validation loss rises. We again state that throughout this training process we are limited by the computational resources of our personal computers, which limits the amount of epochs we can perform as well as adding additional model complexity. In conclusion, our best model produced an overall accuracy of 79%, performing exceptionally well on the normal class with an F1-score of 94%, while having an F1-score of 76% and 68% for the bacteria and virus classes respectively.

We would briefly like to discuss techniques we attempted to break through this ceiling of 79% accuracy.

Learning rate:

- In previous deep learning models, we had learned how important learning rate is for finding an optimal local minimum loss. When we attempted to increase the learning rate, we found that the validation loss would actually increase, probably because the model would be overshooting the loss function and bouncing itself out of local minimums. Alternatively, when attempting to decrease the learning rate we would find that our model wouldn't converge in a reasonable amount of time, hampered by our computational limit.

Regularization:

- Our best model performance came when we implemented regularization in the form of dropout and early stopping. However, when attempting to increase regularization by increasing the dropout percentage, we would find our model would fail to learn the training data properly and struggle with validation.

Image Resolution:

- We achieved our best model performance when decreasing the input image resolution to 128x128, speeding up training time significantly. We believe this worked because we were able to perform more epochs at the cost of decreased performance per epoch, meaning the model could adjust weights more frequently. However, it is safe to assume that with a higher image resolution input the model would be able to detect features in the image more easily and thus result in better performance. This is another case of computational limit.

5. Discussion

While achieving 79% accuracy in training a computer vision model for lung disease classification, it is crucial to emphasize that we do not advocate for this model to serve as the sole arbiter in diagnosing lung problems. Notably, while the model exhibited exceptional performance in classifying normal lung cases, achieving a 99% precision, its performance in the virus and bacteria classes was comparatively lower. This assertion finds support in the findings of Arts L et al., whose study revealed that the diagnostic accuracy of doctors, with specificity at 37% and sensitivity at 89%, demonstrated room for improvement, especially with additional consultations or external opinions.

Our proposition is that computer vision models can offer valuable suggestions to medical professionals but should not be the exclusive basis for diagnosis. Various contextual factors, such as a shortage of medical personnel, may necessitate the integration of AI in decision-making processes. However, the weight assigned to the AI's opinion should be tempered, and decisions should ideally involve human judgment, unless the AI demonstrates an exceptionally high level of confidence (e.g., 99.99% or higher) or aligns with patient comfort levels.

6. REFERENCES

[1] Kadir T, Gleeson F. Lung cancer prediction using machine learning and advanced imaging techniques. Transl Lung Cancer Res. 2018 Jun;7(3):304-312. doi: 10.21037/tlcr.2018.05.15. PMID: 30050768; PMCID: PMC60379

[2] Arts L, Lim EHT, van de Ven PM, Heunks L, Tuinman PR. The diagnostic accuracy of lung auscultation in adult patients with acute pulmonary pathologies: a meta-analysis. Sci Rep. 2020 Apr 30;10(1):7347. doi: 10.1038/s41598-020-64405-6. PMID: 32355210; PMCID: PMC7192898.