CS 470/670 – Introduction to A.I.
Group 6 Final Presentation:

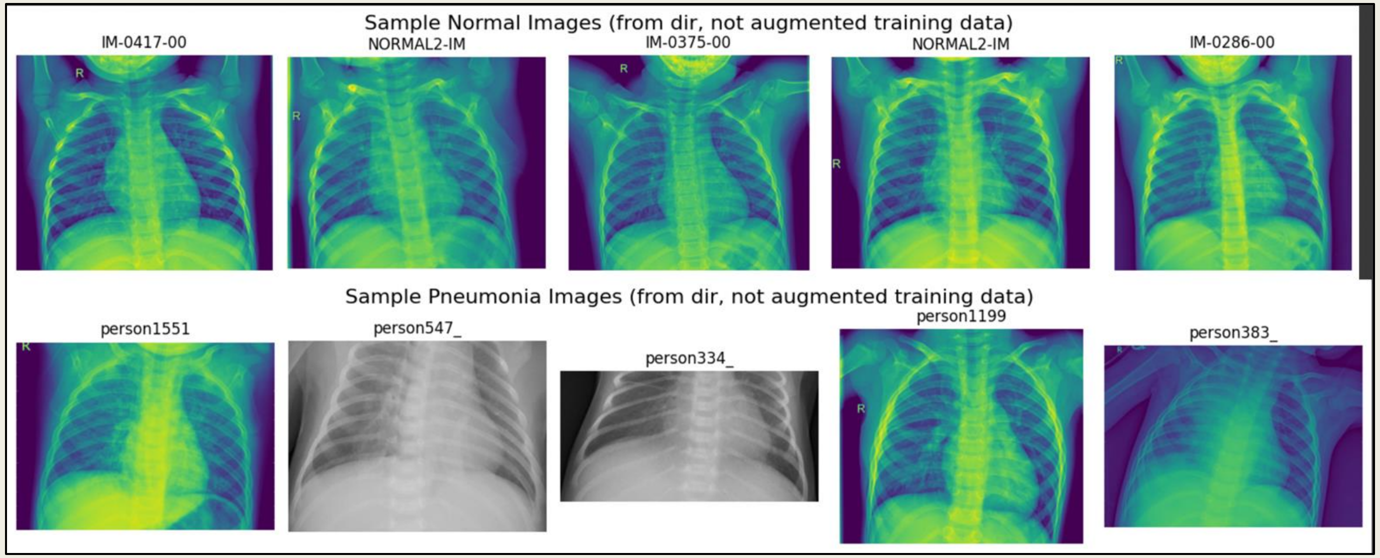
Training Classification Models for Detecting Pneumonia Using X-Ray Images

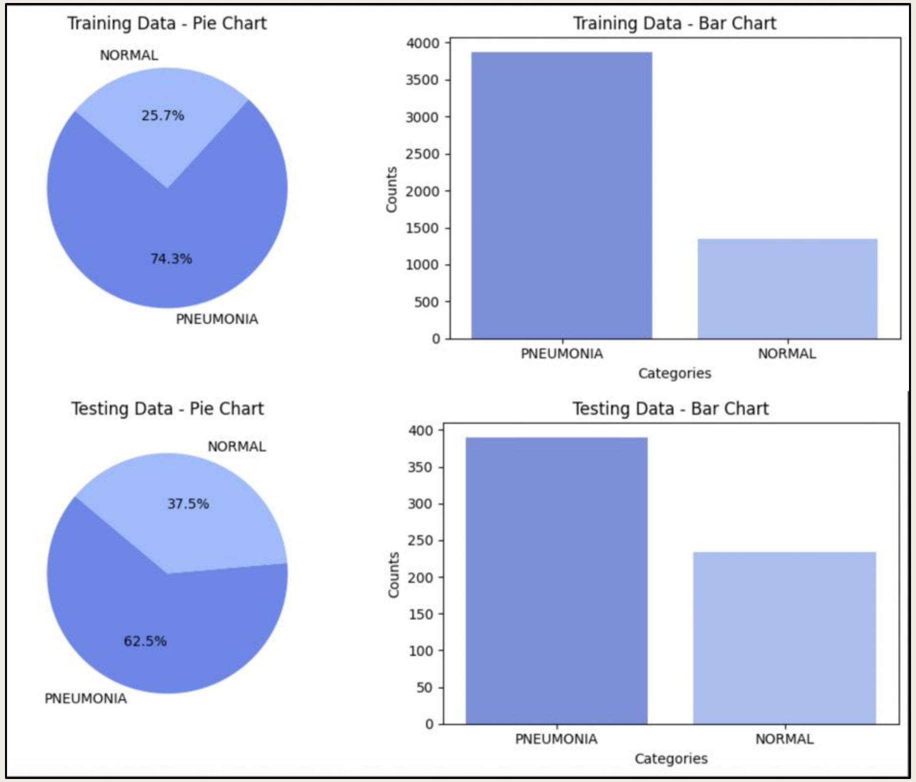
Shabnam Azizi
Ahmed Fassassi
Kaif Khalifa
Maxim Moskalenko
Putha Sri Hari Reddy
Trinadhreddy Seelam

Introduction

<u>Dataset Info</u>

- Kaggle's "Chest X-Ray Images (Pneumonia)" dataset
- 5,863 X-ray images labeled as either:
  - ***"Normal"*** → Healthy patient
  - ***"Pneumonia"*** → Patient with pneumonia
- Unique split:
  - 32% Normal, 68% Pneumonia





Problem Statement

Problem:

- Pneumonia is a very common lung infection with over 3 million US cases per year *(source: Mayo Clinic)*
- Pneumonia is hard to detect with the naked eye through x-ray observation, even for expert radiologists
- Early-stage detection is essential

Goal:

- Build a model that can accurately detect if patients have pneumonia through x-ray images

Use Cases:

- Streamline the screening process for diagnosing patients or determining if they need medical attention
- Provide scalable, cost-effective initial screenings to aid early intervention and alleviate burden on medical professionals.

Brainstorming

<u>Proposed Models:</u>

- ResNet50 models (transfer learning model + fine-tuned model)
- Custom CNN (from scratch with no pretrained weights)
- MobileNetV2 models (transfer learning model + fine-tuned model)

<u>Preprocessing Techniques:</u>

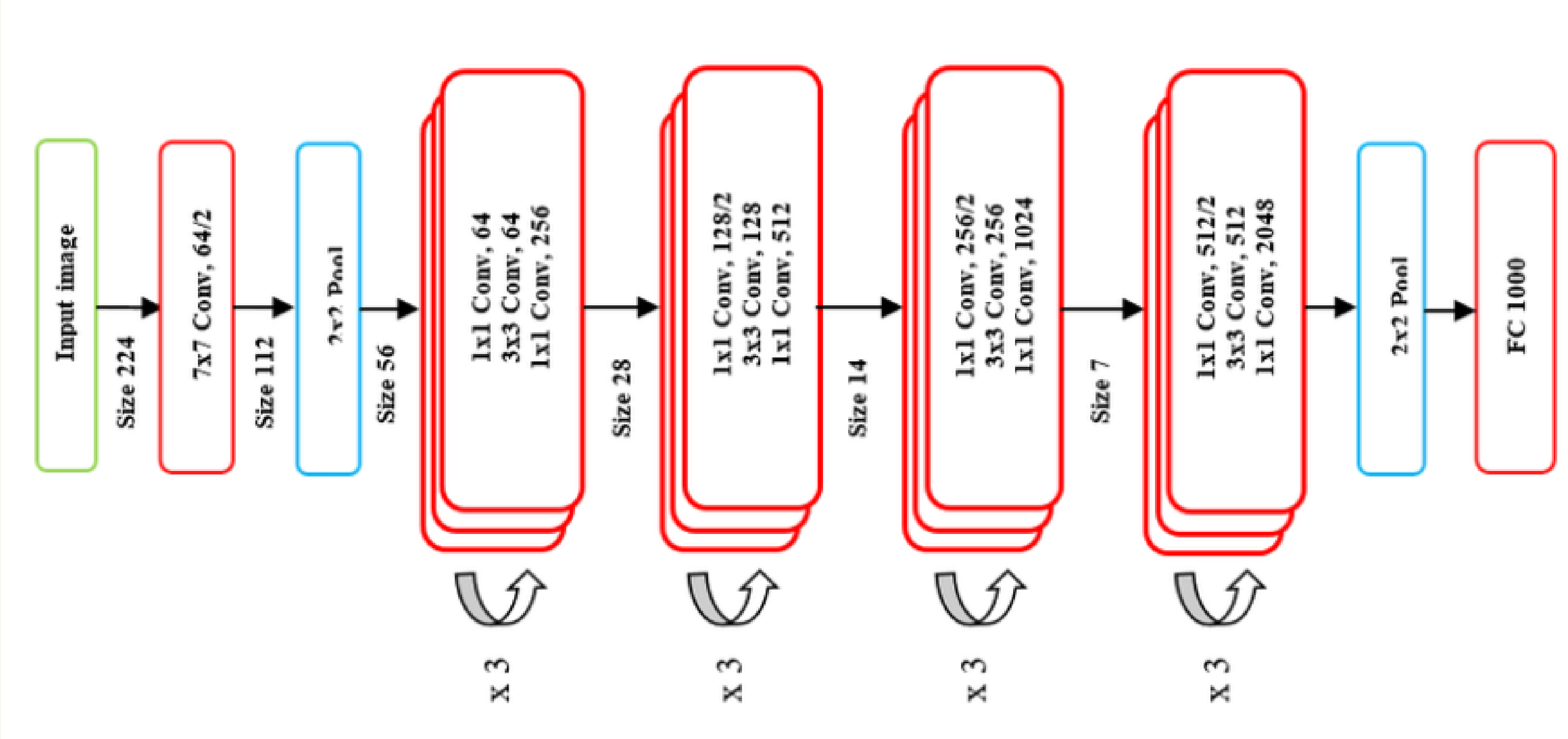
- Phase 1:
  - *Partitioned training, validation and testing data,*
  - *Batched and resized images to uniform scale and resolution*
- Phase 2: Added data augmentation
  - *Random flipping, rescaling, zooming, stretching rotating, etc. to mitigate overfitting during training*

<u>Evaluation Metrics and Visualizations:</u>

- Model accuracy, monitor validation loss for overfitting
- Confusion Matrix visualizations

```
# Create data augmentation object
datagen = ImageDataGenerator(
    rescale=1.0 / 255.0,  # Normalize pixel
    rotation_range=20,    # Rotate images ra
    width_shift_range=0.1,  # Shift horizont
    height_shift_range=0.1,  # Shift vertica
    shear_range=0.1,       # Apply shearing
    zoom_range=0.2,        # Random zoom
    horizontal_flip=True,  # Flip horizonta
    validation_split=0.1   # Reserve 10% of
)
```

Part 1 – Training ResNet50 models
[transfer learning model + finetuned model]



ResNet50 base-model architecture:

Layer info:

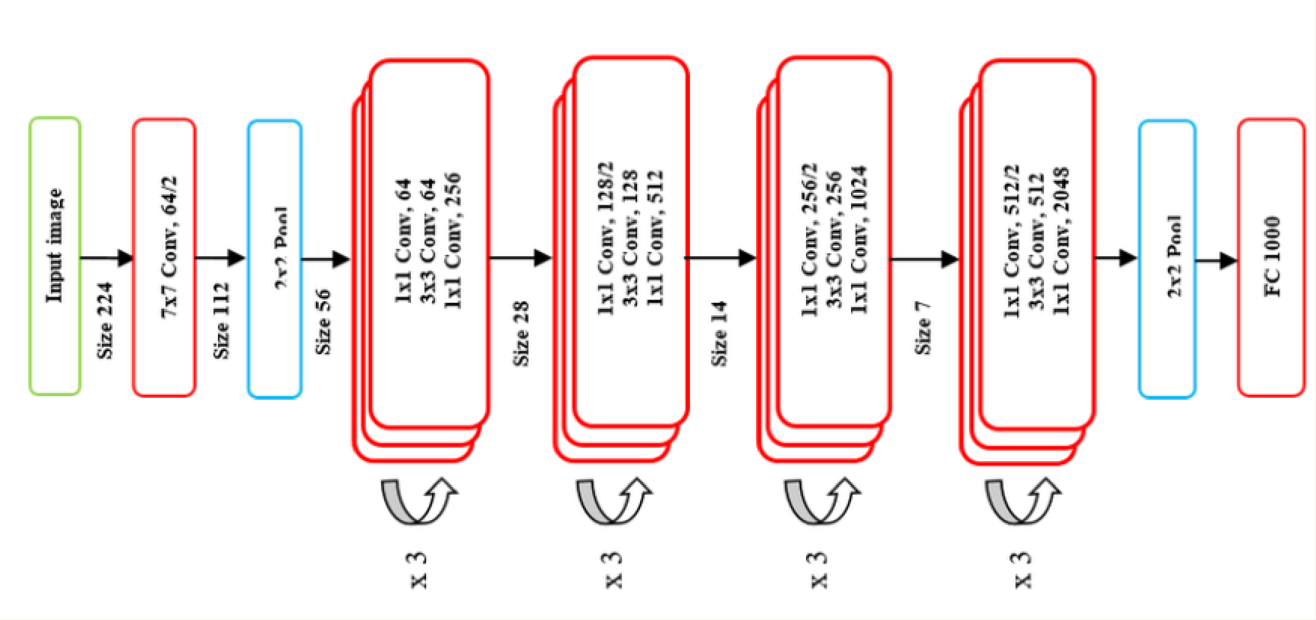
- Widely-known 50-layer CNN
- Complex combination of convolutional layers for precise feature extraction

Pretrained Weights:

- Trained on ImageNet (1M+ labeled images).
- Strong baseline for transfer learning

Applications:

- Applied in medical imaging, object detection, and autonomous systems.
- Great for feature extraction and domain-specific fine-tuning.

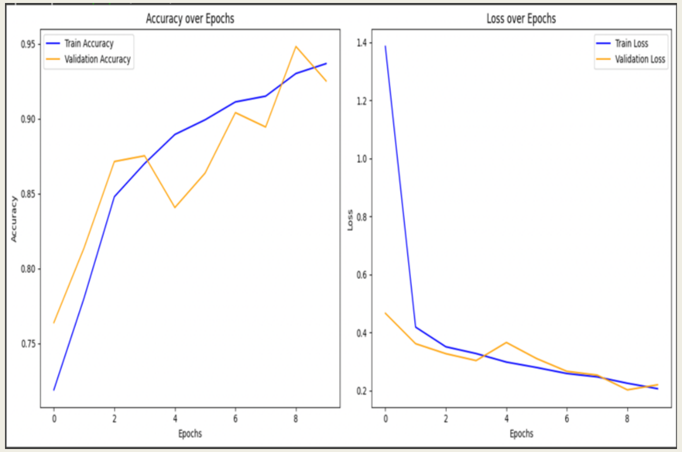


- The 50 layers of the ResNet50 CNN

ResNet50 Model(s) Performance

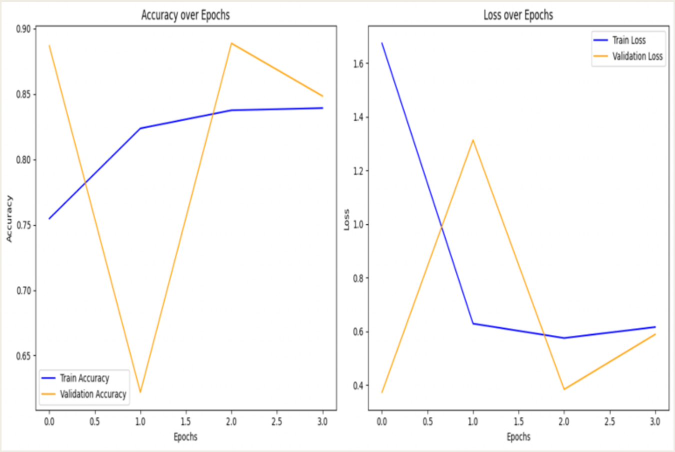
#1 - Transfer Learning Model

- No data-augmentation
- Total epochs trained: 10
- Results:
  - **Accuracy: 76 %**
- Training Graph:



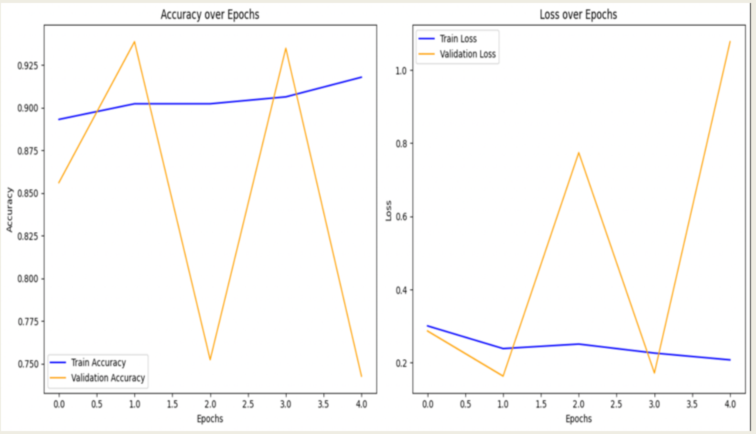
#2 - Transfer Learning Model v2

- With data-augmentation
- Total epochs trained: 10
  - *Also trained for 15, but slightly worsened results (overfitting?)*
- Results:
  - **Accuracy: 84 %**
- Training Graph:



#3 – Finetuned Model

- With data-augmentation
- Total epochs trained: 10
- Results:
  - **Accuracy: 82 %**
  - **(before data-augmentation => 64 %)**
  - Early stopped due to influx in validation accuracy (see below)
- Training Graph:

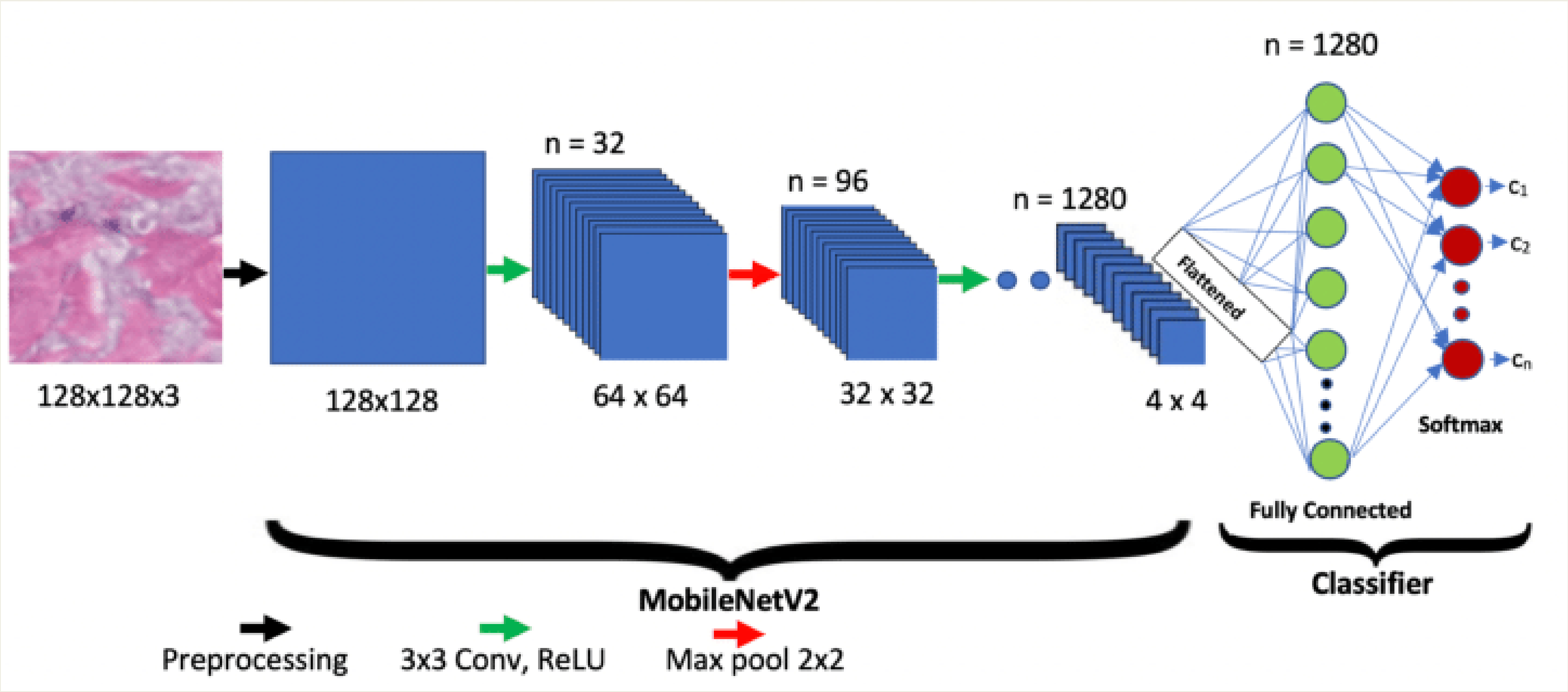


Observations:

ResNet50 is <u>best</u> used with our data set for <u>transfer learning</u>

- Why?
  - <u>Transfer Learning:</u> *“A technique in machine learning in which knowledge learned from a task is re-used in order to boost performance on a related task.”*
  - Due to our relatively small dataset (~4,000 training images), not enough examples to fit the complexity of ResNet50 for fine-tuning (2.3 million parameters, 50 layers)
  - Sources that explore this topic:
    - ResNet50 used for transfer learning with lung X-Ray dataset for detecting COVID-19
    - Transfer learning vs Finetuning (TensorFlow) - https://www.tensorflow.org/tutorials/images/transfer_learning
- Data augmentation <u>significantly</u> increases accuracy

Part 2 –MobileNetV2 transfer learning model + building custom CNN model



Other Models' Architecture & Details

MobileNetV2

- Lightweight CNN with:
  - inverted residual blocks
  - depth-wise separable convolutions
  - optimized for efficient feature extraction and transfer learning
  - Used in [medical applications]()
- Developed by Google Research team in 2018
- Aimed for lightweight system use (i.e. smartphones, IoT devices, embedded systems, etc.)
- Pretrained Weights: Trained on ImageNet (like ResNet50)

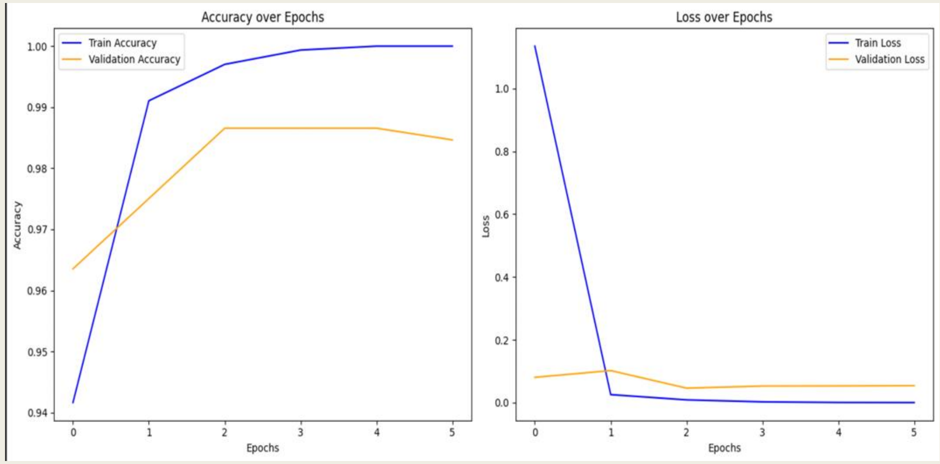
Custom CNN

Layer (type)	Output Shape	Param #
conv2d_9 (Conv2D)	(None, 254, 254, 32)	896
batch_normalization_9 (BatchNormalization)	(None, 254, 254, 32)	128
max_pooling2d_9 (MaxPooling2D)	(None, 127, 127, 32)	0
dropout_12 (Dropout)	(None, 127, 127, 32)	0
conv2d_10 (Conv2D)	(None, 125, 125, 64)	18,496
batch_normalization_10 (BatchNormalization)	(None, 125, 125, 64)	256
max_pooling2d_10 (MaxPooling2D)	(None, 62, 62, 64)	0
dropout_13 (Dropout)	(None, 62, 62, 64)	0
conv2d_11 (Conv2D)	(None, 60, 60, 128)	73,856
batch_normalization_11 (BatchNormalization)	(None, 60, 60, 128)	512
max_pooling2d_11 (MaxPooling2D)	(None, 30, 30, 128)	0
dropout_14 (Dropout)	(None, 30, 30, 128)	0
flatten_3 (Flatten)	(None, 115200)	0
dense_6 (Dense)	(None, 128)	14,745,728
dropout_15 (Dropout)	(None, 128)	0
dense_7 (Dense)	(None, 1)	129

Other Models' Results

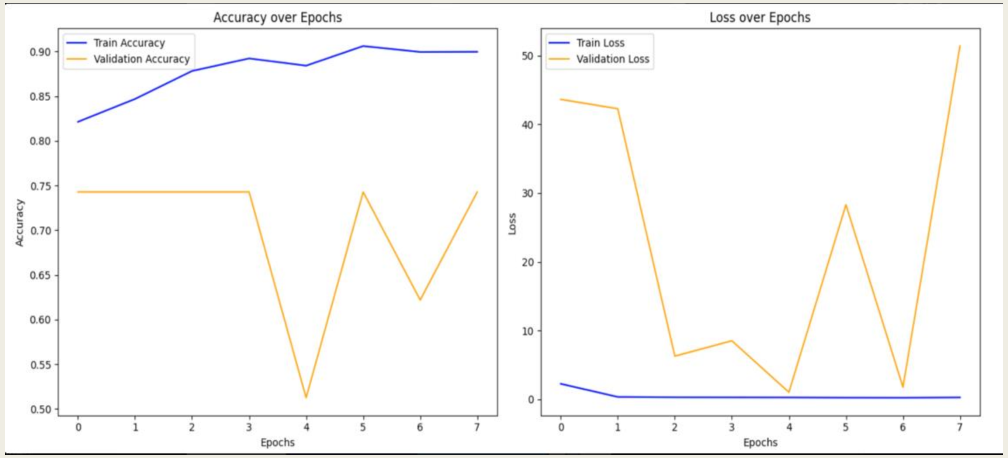
MobileNetV2

- Epochs trained:
  - *Early stopped at 6/15*
- **Accuracy:** 80 %
- Training graph:



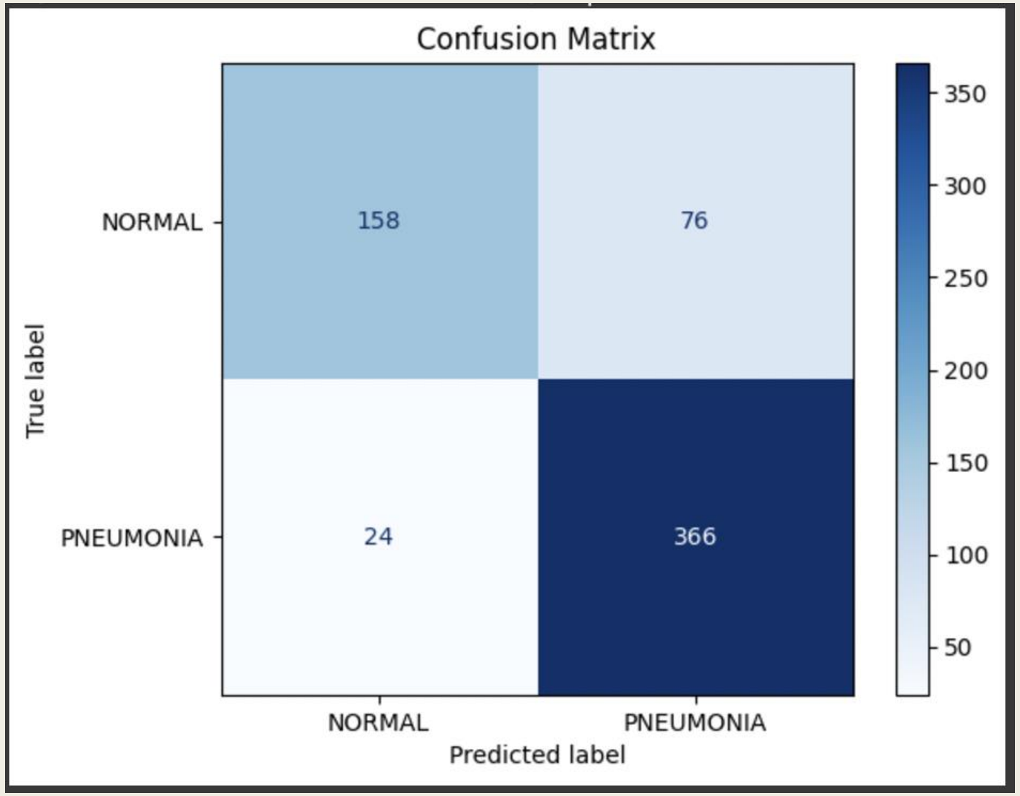
Custom CNN

- Epochs trained:
  - *Early stopped at 7/15*
- **Accuracy:** 61 %
- Training graph:



Observations:

- Poor CNN performance, most likely due to lack of complexity and pretrained weights, compared to MobileNetV2 and ResNet50 models
- So far, our accuracy has plateaued **~80 %**
  - All models are <u>better</u> at predicting which patients <u>have pneumonia</u>, vs which patients <u>don't</u> have pneumonia (normal cases)
    - Due to awkward dataset split?
    - ***[3,800 'Pneumonia' examples vs 1,300 'Normal' examples in training set]***
- We experiment with changes to address this caveat in next section...



ResNet50 transfer learning model Confusion Matrix (overall accuracy => ~84%)

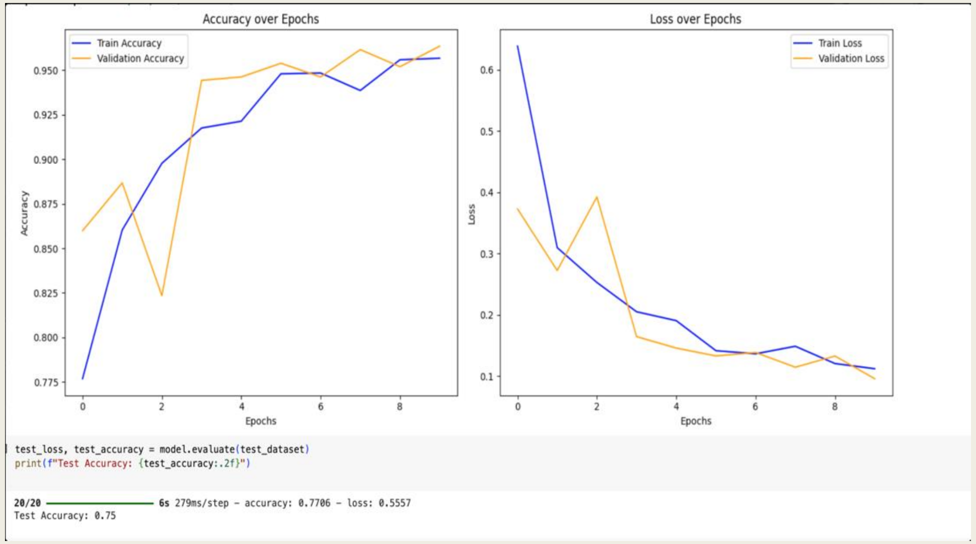
Model accuracy on 'Pneumonia' cases => 93.85 %
Model accuracy on 'Normal' cases => 67.52 %

Part 3 - Attempted Improvements

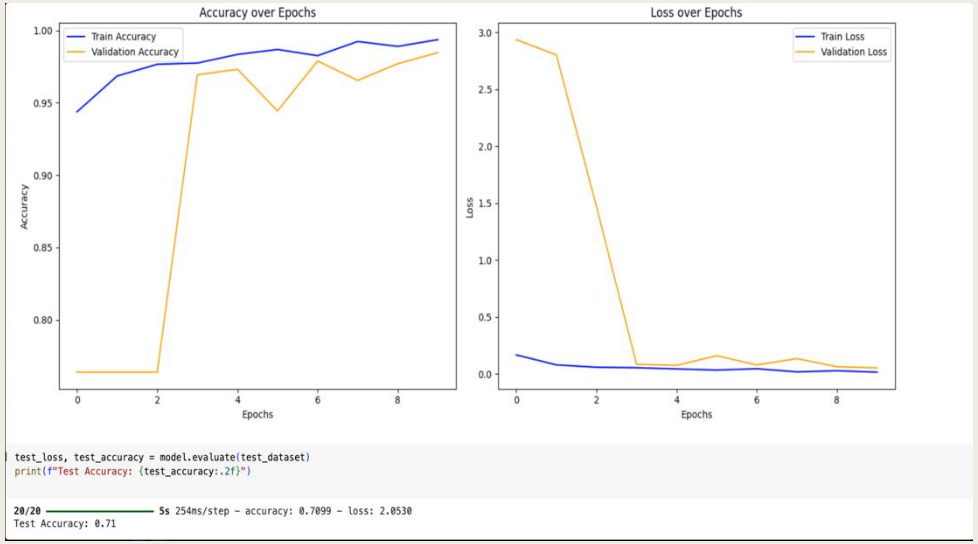
In order to address the unevenness of the dataset split, we experimented with using an equal amount of ‘PNEUMONIA’ and ‘NORMAL’ examples in the training dataset.

Equal Split Dataset Results + Observations

- However, despite the equal dataset split, the two best performing model architectures on the previous data set split underperformed with the new split
- 75 % for transfer learning ResNet50 model
- 71 % for finetuned ResNet50 model
- Early stopping and data augmentation for both models

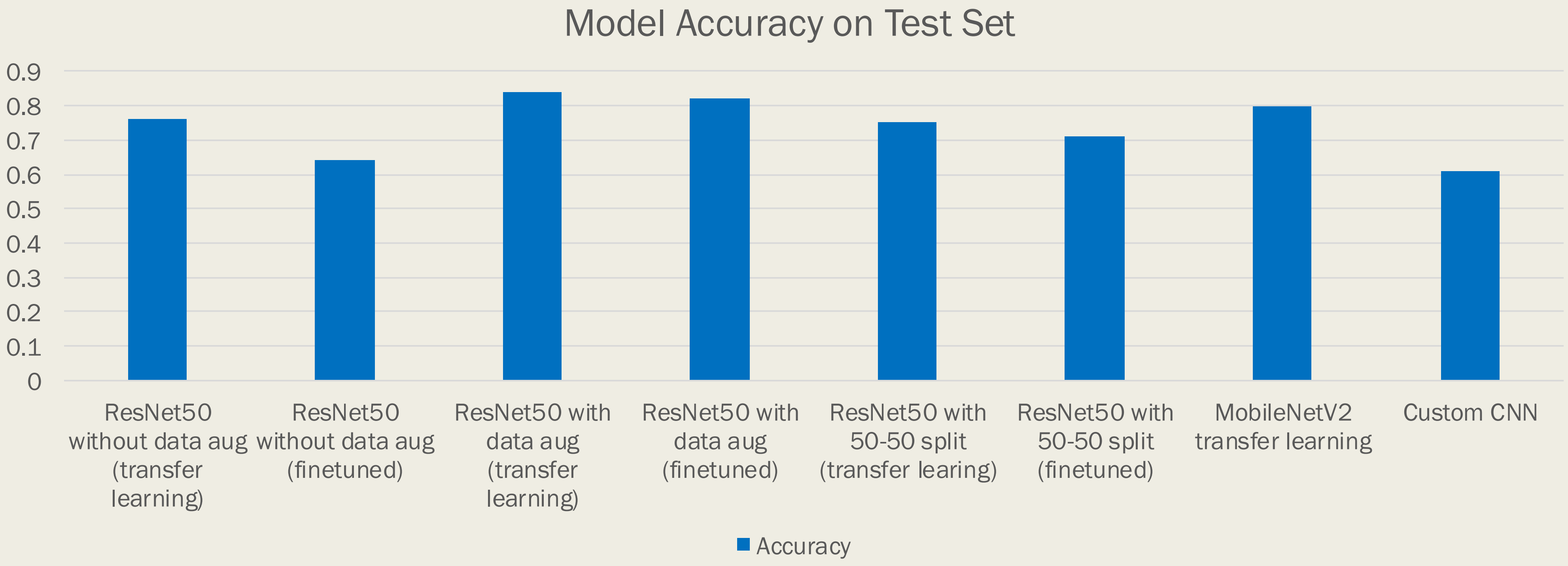


- ResNet50 transfer learning model



- ResNet50 finetuned model

Final Definitive Model Comparison



Conclusion and takeaways

- Best performing model
  - ***ResNet50 transfer learning model with data augmentation => 84 % accuracy on test set***
- At a highest accuracy of 84%, we conclude that our model would NOT be suitable for widespread use in the medical field without supervision/second (human) opinion.
  - *Numbers could most likely be improved with larger, more balanced dataset*
- Transfer learning with a complex model pretrained on a large, robust dataset (ImageNet) for relatively small dataset (our ~6000 image dataset) is more effective than fine-tuning, in terms of feature extraction and image classification tasks.
- Data augmentation is a great way to prevent overfitting and improve model accuracy, given the conditions of our dataset.

Sources

- Akay, Metin et all. *[Deep Learning Classification of Systemic Sclerosis Skin Using the MobileNetV2 Model]* - *https://www.researchgate.net/publication/350152088_Deep_Learning_Classification_of_Systemic_Sclerosis_Skin_Using_the_MobileNetV2_Model*
- *Mark Sandler et all., MobileNetV2: Inverted Residuals and Linear Bottlenecks* - *https://arxiv.org/abs/1801.04381*
- *Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images* - https://www.sciencedirect.com/science/article/pii/S235291482200065X
- Wikipedia – Transfer Learning - https://en.wikipedia.org/wiki/Transfer_learning

Thank you!

Questions?