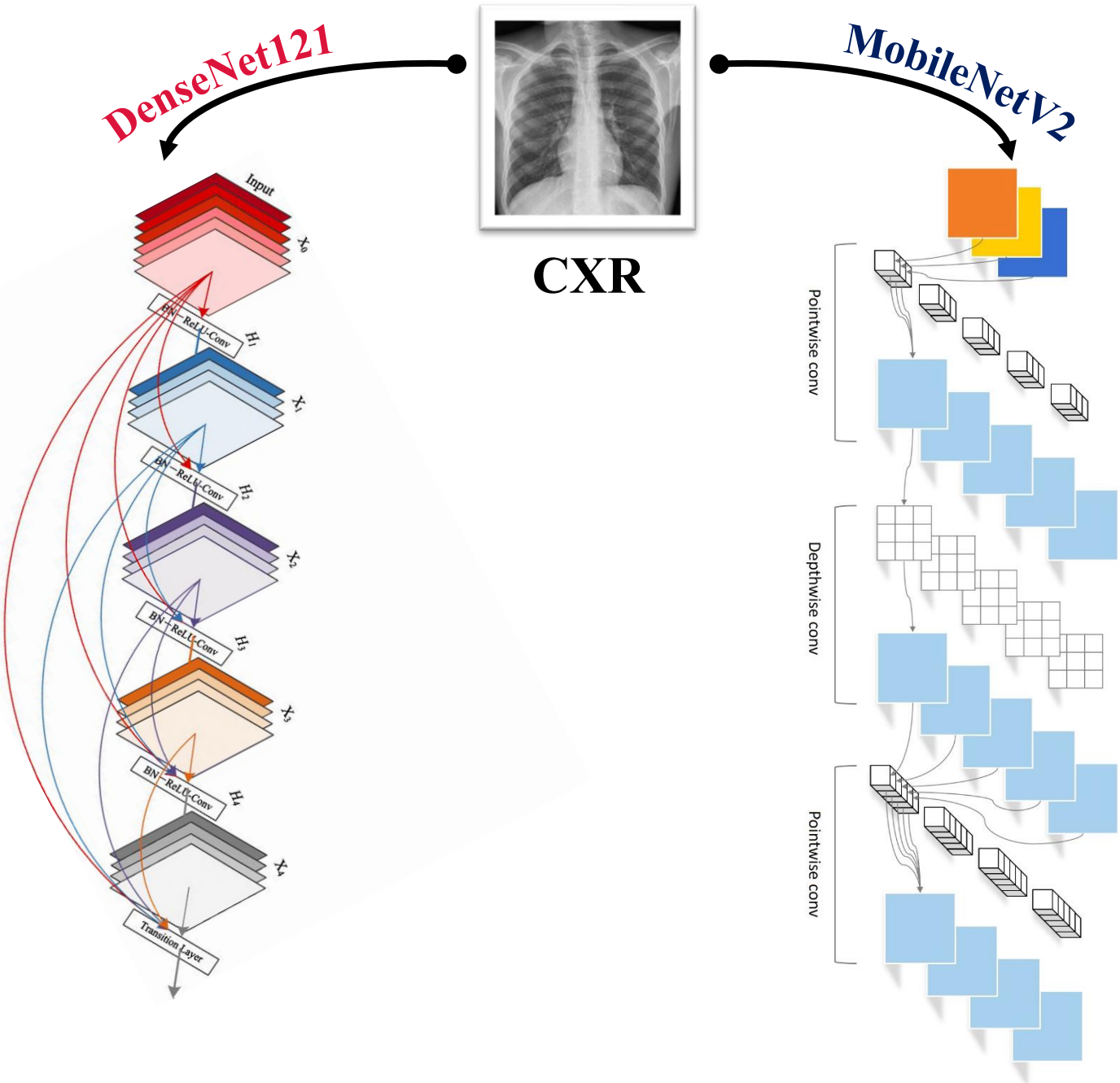
Jordan University of Science and Technology

AI481 - AI for Medicine and Healthcare

Assignment 2:

CXR Classification using MobileNetV2 and DenseNet121



Supervised By: Dr. Ahmad Alzubi

Team Members

- **Hamza** Sami Mohammad **Alsawaftah**
  - **ID: 161395**
- **Nidal** Khaled Abdel Hameed **Shahin**
  - **ID: 162278**

Abstract

In this assignment, we explore the application of **CNN-based** architectures, specifically: **MobileNetV2** and **DenseNet121**, to classify chest X-ray (CXR) images into three categories: **Normal**, **COVID-19**, and **Pneumonia**. Models were trained from scratch without pre-trained weights, identical training configurations and settings applied were adapted from the proposed reference of AlexNet implementation.

Introduction

In respiratory diseases diagnosis, **CXR imaging** is **vital**. Machine and Deep Learning models, **Convolutional Neural Networks** (CNNs) in particular, were the core of automating these diagnoses. In this study, famous architectures like **MobileNetV2** and **DenseNet121** were used to classify chest X-ray images, following guidelines established in the provided GitHub AlexNet notebook **[1]**.

Method

- **Dataset**: Used **COVID-QU-Ex** dataset [2], balanced distribution and richer in images, compared to the proposed alternatives.
- **Models**: MobileNetV2 and DenseNet121, initialized with "`weights=None`", fine-tuned their final layers for three output classes.
- **Preprocessing**:
    - Images were **converted** from **grayscale** to **RGB**.
    - **Resized** to **(224x224)** to match the models' architectures.
    - **Normalized** images using dataset-specific computed **mean** and **standard deviation** (std).
- **Training Settings**:
    - **Optimizer**: **Adam**
    - **Learning rate**: **0.0001**
    - **Batch size**: **32**
    - **Epochs**: **10**
    - **Loss function**: **Cross Entropy Loss**

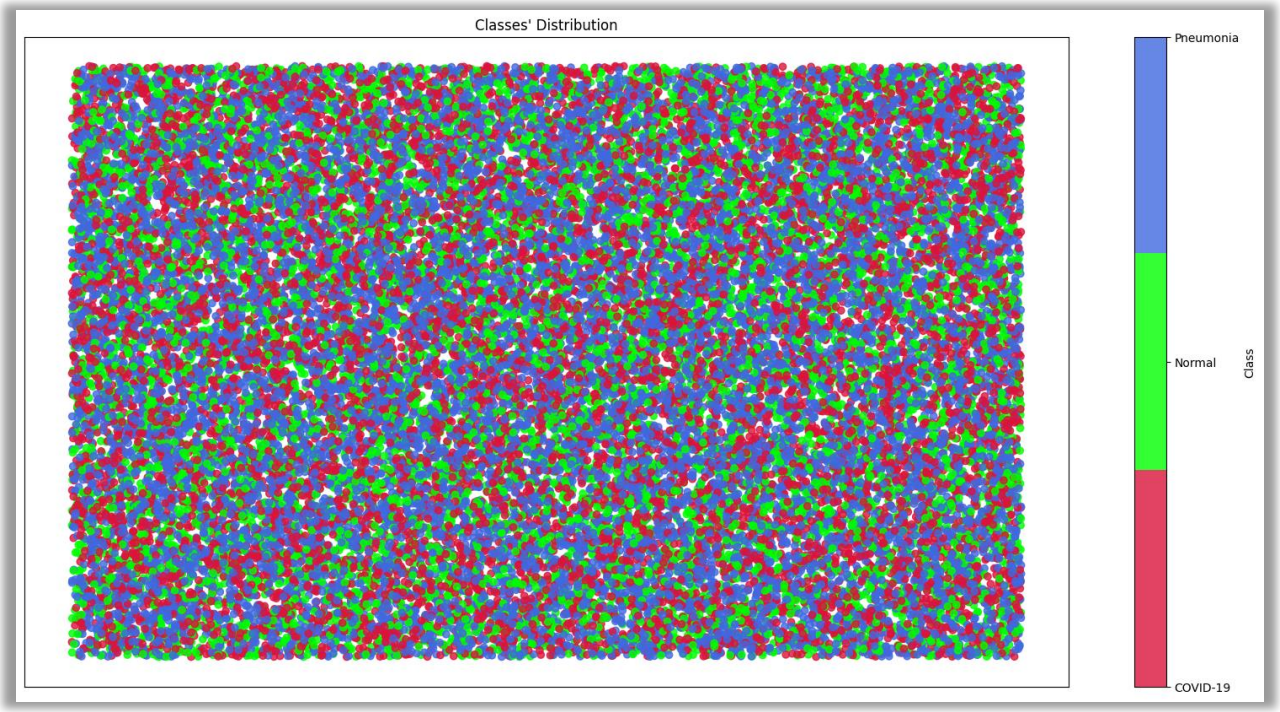
Experiment

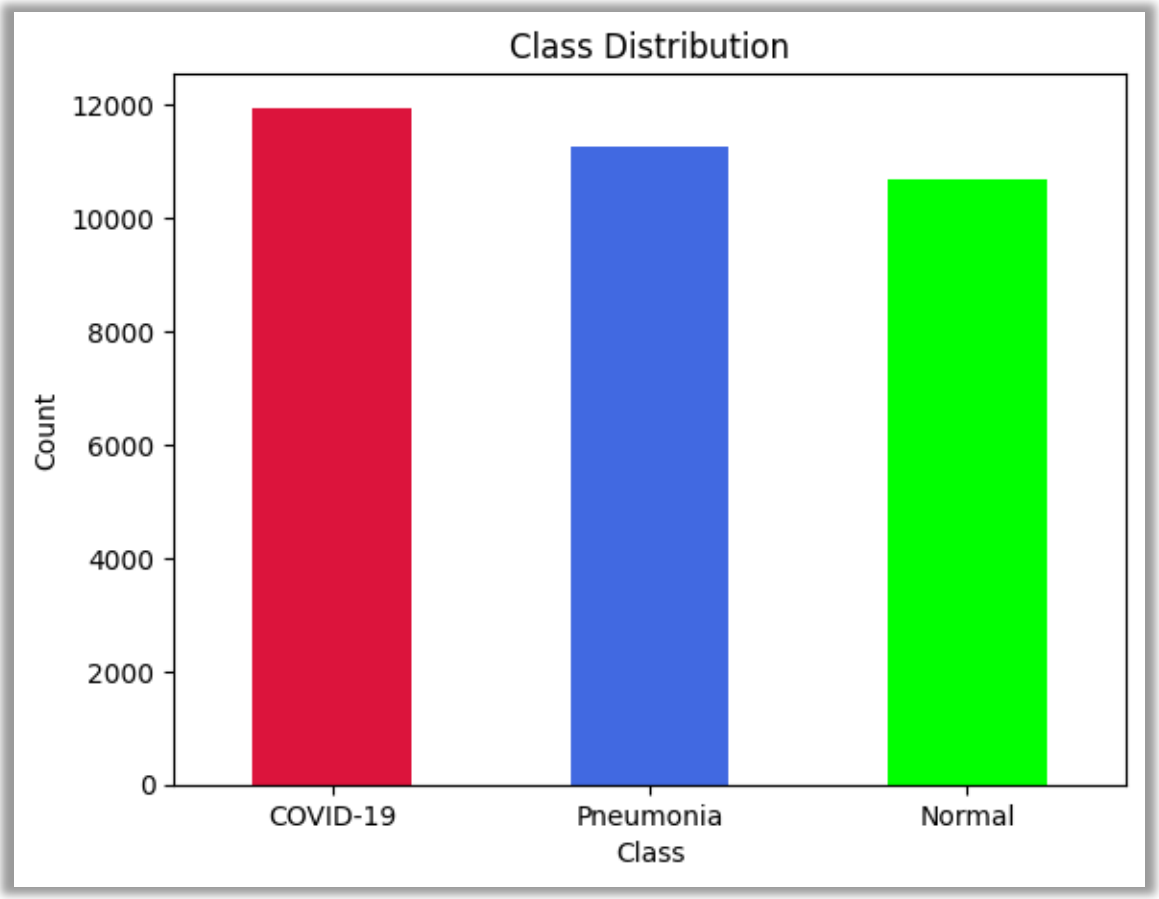
Trained both models **from scratch** using the configuration described above. After each epoch in training, **Validation** was performed to monitor the F1-Score and **save** the model with the best performance.

Evaluation (Testing) was done on a separate specialized **test set**, reporting both **macro** and **per-class** precision, recall, F1-Score, and accuracy for each model. Generated **plots** to compare training & validation loss, accuracy, and F1-Score trends. Confusion matrices and classification reports were also visualized.

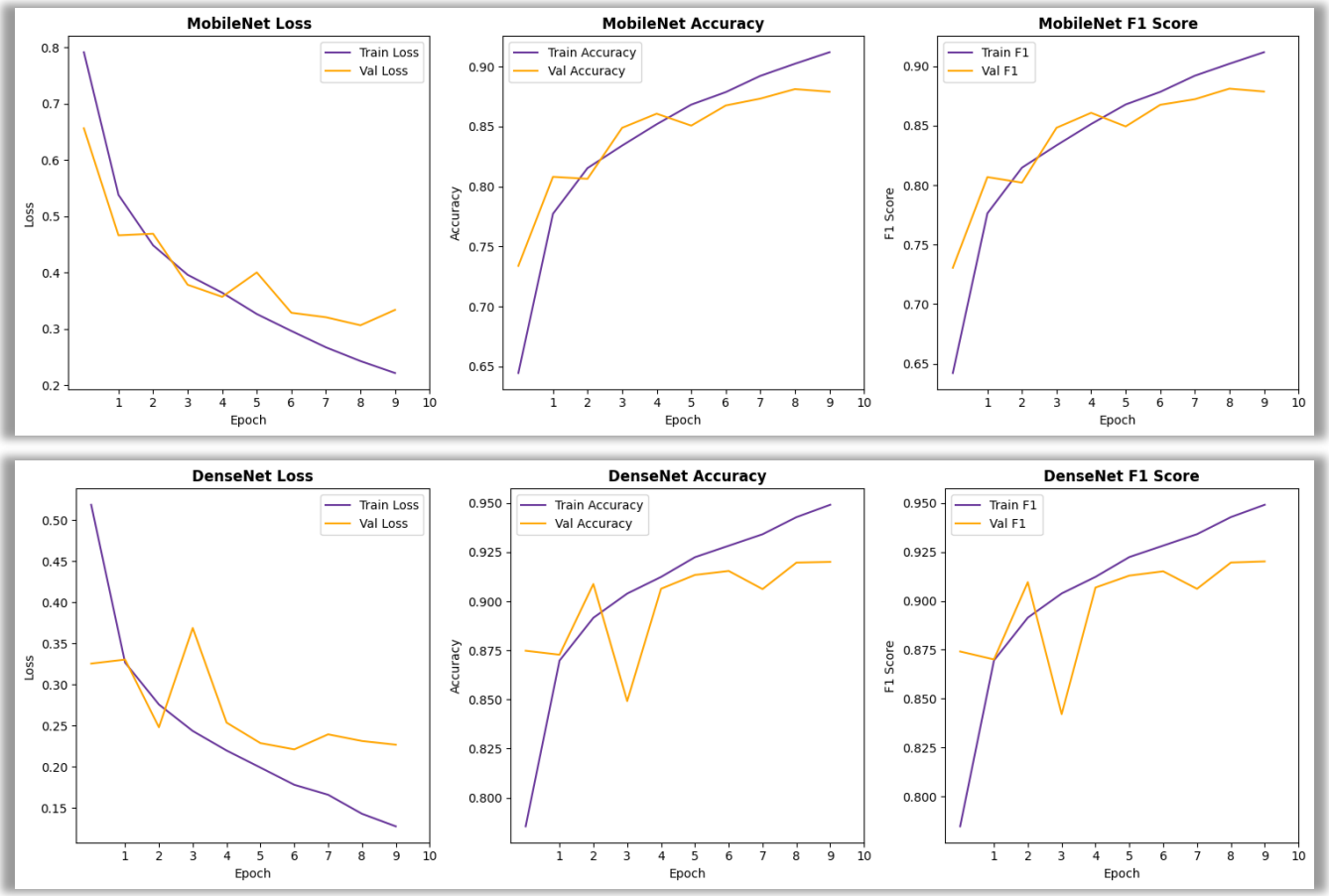
Results & Visuals

1st| Data distribution (class-wise) [3]

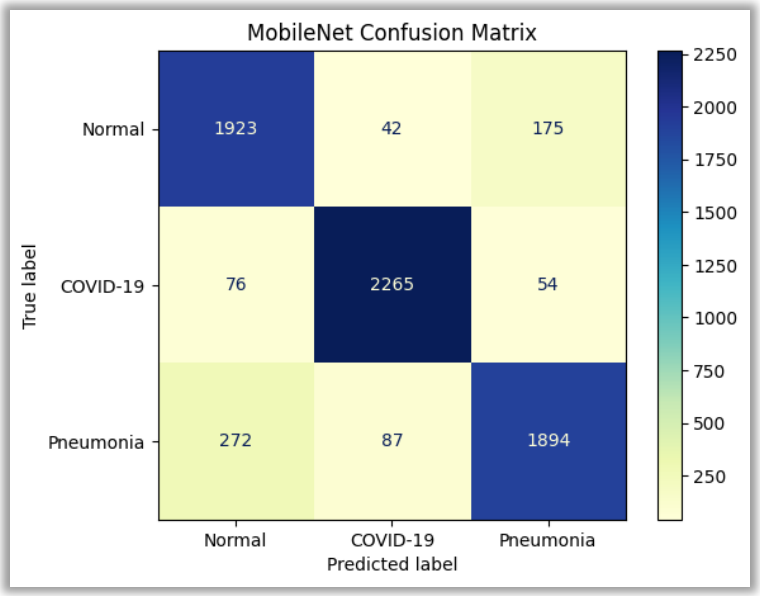


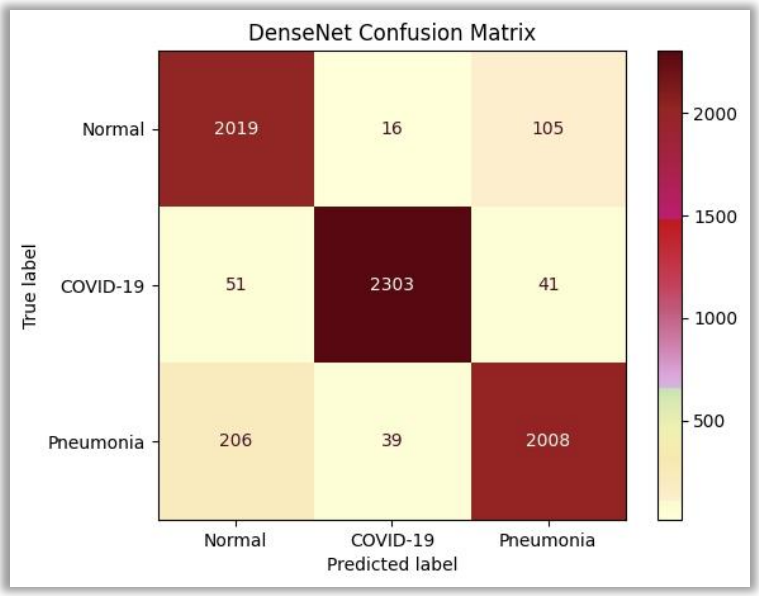


2nd| Training and Validation Trends [3]

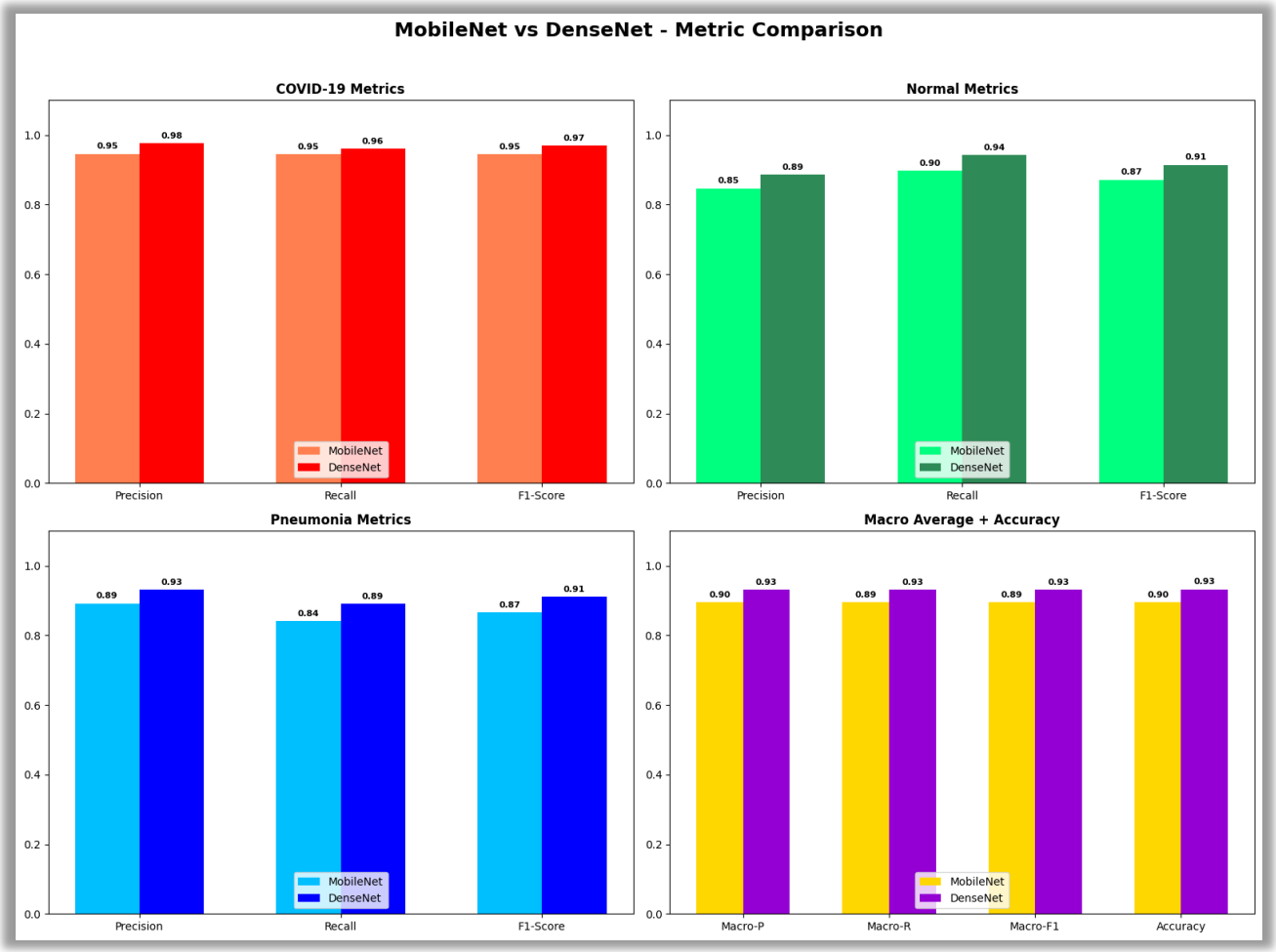


3rd| Evaluation Confusion Matrices [3]





4th| Evaluation Metrics Visual [3]



5th| Evaluation Metrics as a Table

Model	Class	Precision	Recall	F1	Accuracy
MobileNetV2	Macro Avg	0.90	0.89	0.89	0.90
	Normal	0.85	0.90	0.87	-
	COVID-19	0.95	0.95	0.95	-
	Pneumonia	0.89	0.84	0.87	-
DenseNet121	Macro Avg	0.93	0.93	0.93	0.93
	Normal	0.89	0.94	0.91	-
	COVID-19	0.98	0.96	0.97	-
	Pneumonia	0.93	0.89	0.91	-

Discussion

MobileNetV2 and **DenseNet121** both were effective for CXR classification under this constrained setup **[1]**. **DenseNet's** with its **deeper** architecture captured finer & more complex details, thus had **better F1-scores**, especially for **COVID-19** cases. **MobileNetV2**, being **lightweight**, had **3x faster** training time **but** slightly **lower** predictive performance.

Batch size (32) and epochs (10) ensured a feasible computation without sacrificing the models' abilities.

Conclusion

Even when trained from **scratch**, both **MobileNetV2** and **DenseNet121** showed **promising results** for the detection of respiratory disease under the given configuration. **DenseNet121** was the **superior** model in terms of overall **classification performance**, **MobileNetV2** also demonstrated strong potential for supporting medical decisions.

References

[1] *Roshan Sadath, "Medical-Image-Diagnosis-using-Convolutional-Neural-Networks"* **(GitHub Link)**

[2] *COVID-QU-Ex* **(dataset source link)**

[3] Notebook (**Google Collaboratory**), Run-Ready with all requirements **(Kaggle Environment)**