Age and Gender Prediction using CNN

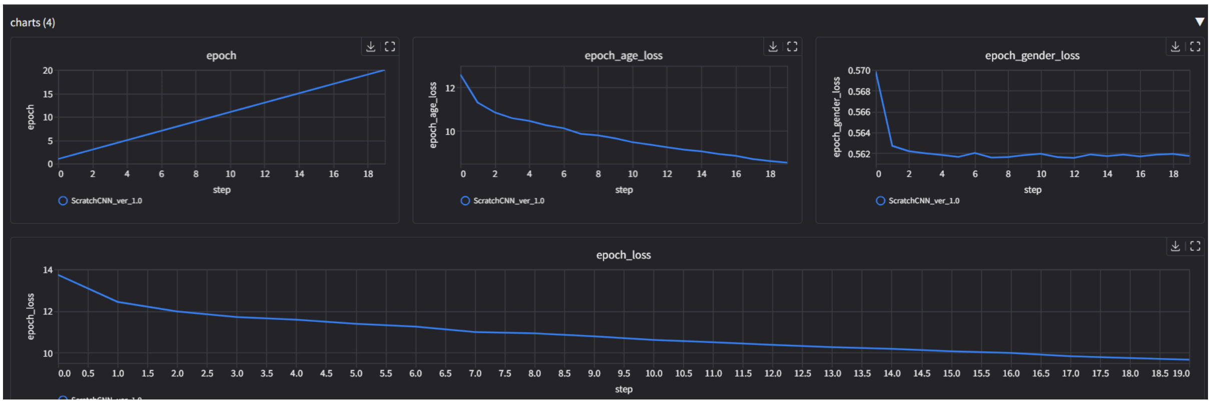
Project Report

The goal of this project is to develop and evaluate deep learning models that can predict gender (classification) and age (regression) from facial images. Two approaches were implemented using PyTorch a Scratch CNN and a Fine-tuned Pretrained Model to compare performance and understand the impact of transfer learning.

• Approach 1: Scratch CNN

In the first approach, a Convolutional Neural Network was designed and trained from scratch.

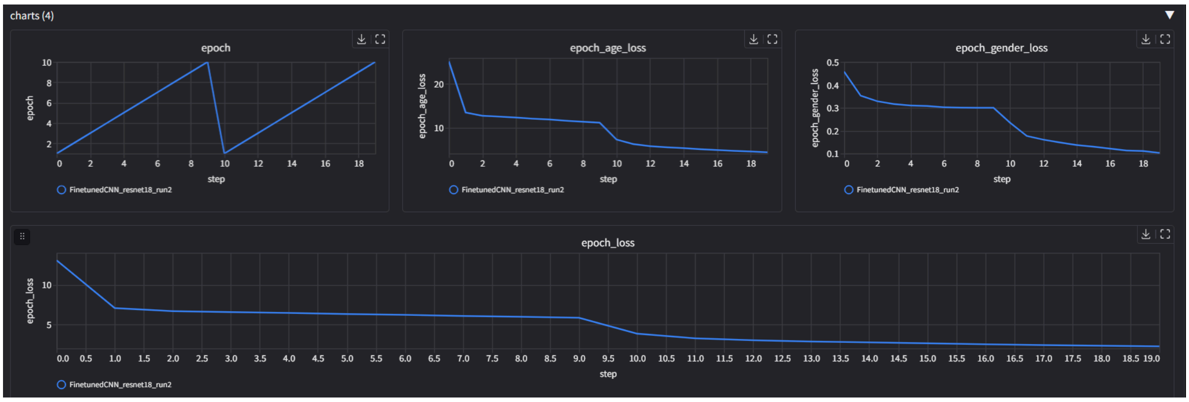
- Architecture: Three convolutional blocks (Conv2D -> BatchNorm -> ReLU -> MaxPool) followed by fully connected layers for two outputs gender (2 classes) and age (continuous).
- Loss Functions:
  - Gender : CrossEntropyLoss()
  - Age : L1Loss()
- Optimizer: Adam (lr = 1e-4)
- Data Augmentation: Random horizontal flips, rotations, and normalization using torchvision.transforms.
- Result: Achieved a public leaderboard score of 0.509, learning basic facial representations but limited by the shallow architecture and fewer learned features.



• Approach 2: Fine-Tuned CNN (ResNet18)

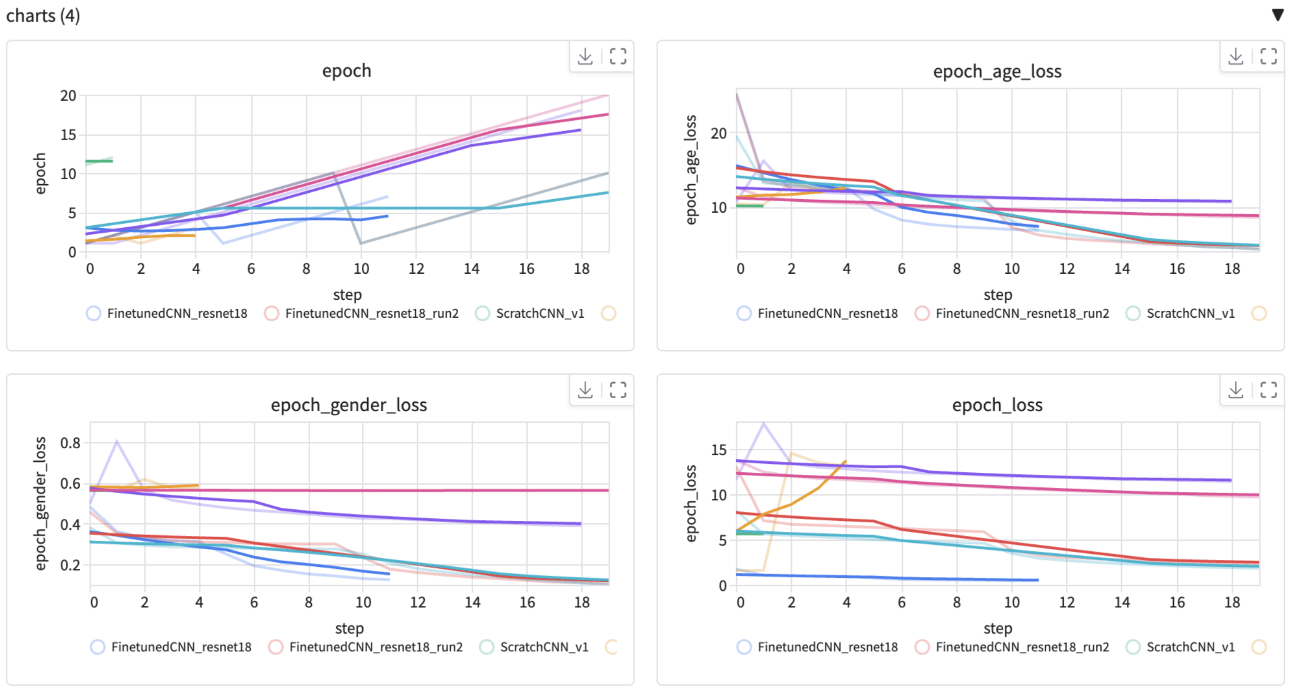
The second approach utilized ResNet18 pretrained on ImageNet, fine-tuned for both gender and age prediction.

- Modification: Replaced the final fully connected layer with two output heads one for gender classification and one for age regression.
- Training Strategy:
  - Initially froze the pretrained backbone and trained only the heads for 10 epochs (lr = 1e-4).
  - Then unfroze all layers and fine-tuned for additional 10 epochs (lr = 1e-5).
- Loss Functions: Same as the scratch model — combined CrossEntropyLoss and L1Loss.
- Result: Achieved a public leaderboard score of 0.8111, showing a major improvement due to transfer learning and robust pretrained feature extraction.



- **Comparison and Results:**

Model	Architecture	Loss Function	Training Epochs	Public Score	Remarks
Scratch CNN	Custom CNN	CE + L1	20	0.509	Baseline model
Fine-Tuned CNN	ResNet18 (Pretrained)	CE + L1	20	0.8111	Significantly better due to transfer learning



- **Resources Reffered :**

  Book : Deep Learning with PyTorch Book by Eli Stevens, Luca Antiga, and Thomas Viehmann.
  YT : click here