CNN fine-tuning for age prediction

Simone Vaccari 915222
Davide Vettore 868855

1. Objective:

The goal of this experiment is to perform age prediction on the IMDB-WIKI dataset by applying transfer learning of different pre-trained Convolutional Neural Networks. Afterwards, we'll assess the performance of these models by comparing the results using two different correlation coefficients.

2. Description of the dataset:

The IMDB-WIKI dataset consists of 523,051 images of celebrities sourced from IMDb and Wikipedia. For our task we initially selected only the 460,723 images extracted from IMDb, and then removed the ones that did not contain any face or contained more than one. Additionally, we excluded faces with a *face_score* falling below a certain threshold and those indicating ages outside the 0-100 years interval. After this initial step, we were left with about 72,000 samples, half of which were selected to compose the final dataset. These remaining samples were then split into training set (21,568 images), validation set (7,183 images), and testing set (7,183 images).

Both the training and the validation images were resized to 256x256, and normalized. Moreover, in order to perform data augmentation, we randomly applied horizontal flipping to the training samples with a probability of 0.5. Figure 1 shows some random training samples before applying these transformations, while Figure 2 displays the corresponding faces after this initial processing phase.

Figure 1: Examples of training faces before normalization

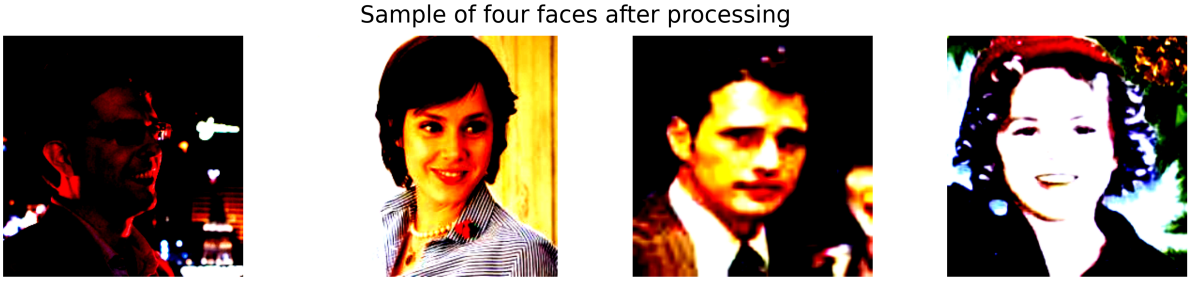

Figure 2: Examples of training faces after normalization

The dataset is composed of multiple variables that were only used in the pre-processing steps. Variables like *face_score*, *second_face_score*, *date_of_birth* and *photo_taken* (those last two are converted into *age*) can now be removed. In Figure 3, we can observe that the age distribution remains quite consistent across the different subsets obtained after completing the split.

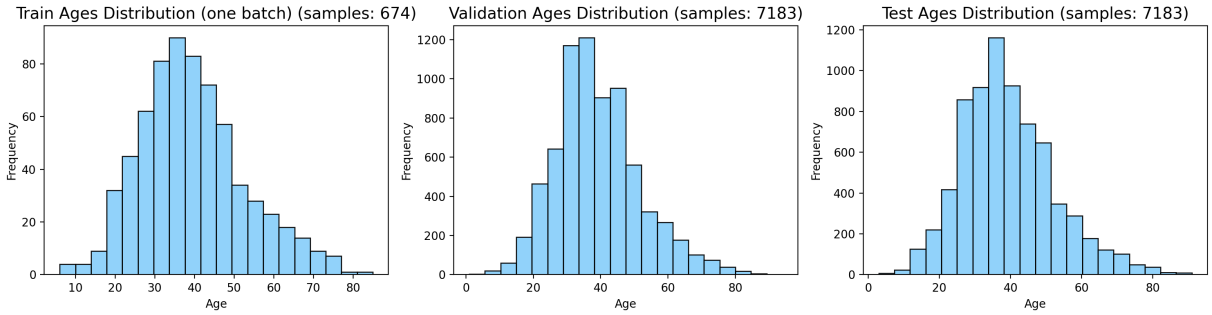

Figure 3: Age distribution in the subsets

3. Procedure:

In order to perform our age regression task we used the concept of **transfer learning**: this technique exploits the fact that earlier layers of a CNN contain more generic features that can be used for many tasks, while features in later layers become more and more specific to the details of the classes in the original dataset. With this in mind, the main idea of transfer learning is to transfer the knowledge of the intial layers of a pre-trained network to a new CNN that presents differences in the latest layers. This operation is done by cutting the original network at a certain location, append the new layers, optionally freeze the initial ones, and update the parameters only in the unfrozen layers. The method just described is known as **fine-tuning**.

In our case, we used two different networks, namely **MobileNet_V2** and **EfficientNet_B3**, both pre-trained on the *ImageNet-1K* dataset, containing more than 1 million images of 1,000 different classes. In both cases, the final predictor of the network was replaced with three linear layers properly defined, with GeLU activation function.

Table 1 reports the Top-1 and Top-5 accuracies of the two models on the *ImageNet-1K* dataset, as well as the total number of trainable parameters, and the number of floating-point operations.

Network	Acc@1	Acc@5	Params	GFLOPS
MobileNet_V2	71.878	90.286	3.5M	0.3
EfficientNet_B3	82.008	96.054	12.2M	1.83

Table 1: Characteristics of the used architectures

After defining the model architecture, we operated in two different ways:

1. By freezing all layers but the linear ones that were added at the end;
2. By keeping all layers unfrozen;

In the first case, only the weights of the last three layers were updated during training, while in the second one the whole network was fine-tuned. In order to evaluate the progress of the training phase, conducted with the Adam optimizer and a decreasing learning rate, a validation was performed at the end of each epoch, using the mean absolute error as loss function.

Once the training was completed, we tested the performances of the models on the testing set by computing the correlation between the predicted ages and the ground-truths. We employed two different correlation coefficients:

- **Pearson Linear Correlation Coefficient** (PLCC), which measures the linear relationship between two variables;
- **Spearman Rank-Order Correlation Coefficient** (SROCC), a nonparametric measure of the monotonicity of the relationship between two variables;

Both coefficients vary between −1 and +1, with 0 implying no correlation, and −1 or +1 implying an exact linear relationship.

4. Results:

The results obtained by the different networks are reported in Table 2. It can be noted that the performance of both models considerably improves when transitioning from the partial fine-tuning to the full fine-tuning. As anticipated, the network architecture with a higher number of parameters and computational requirements performed better in both approaches.

Network	Test loss	PLCC	SROCC
MobileNet_V2 Linear layers fine-tuned	0.076	0.658	0.592
MobileNet_V2 All layers fine-tuned	0.052	0.842	0.823
EfficientNet_B3 Linear layers fine-tuned	0.087	0.678	0.620
EfficientNet_B3 All layers fine-tuned	**0.047**	**0.868**	**0.852**

Table 2: Performances of the different networks

Figure 4 shows a scatterplot of the predicted ages versus the ground-truths, for the MobileNet and the EfficientNet respectively. The dark blue dots represent the samples for which the difference between real and predicted age is greater than 40 years, while the blue line illustrates a reference linear regression among all the points.

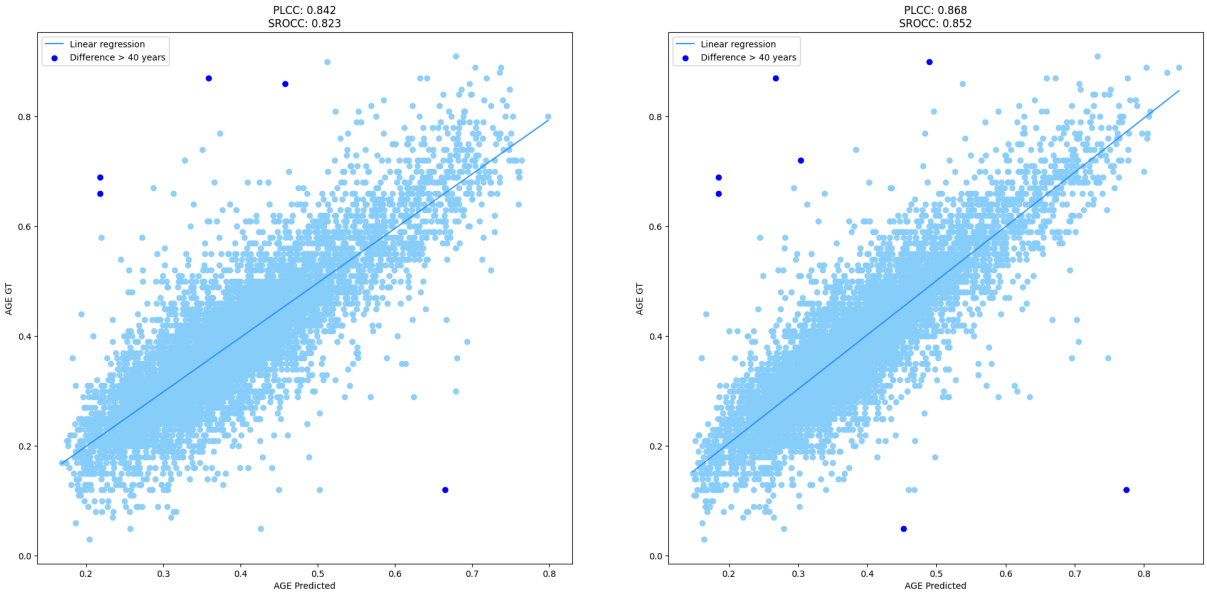

Figure 4: Scatterplot of the predicted ages versus the ground-truths for the fully fine-tuned MobileNet and EfficientNet respectively

In Figure 5 and Figure 6 we presented the misclassified instances highlighted in the two scatterplots (prediction error > 40 years). Upon observing these alarming misclassification, we notice that two major problems occur. Both appear to be associated with the training set rather than the model itself.

Firstly, by looking at the displayed faces, it seems improbable that the real age is accurate, while the predicted age seems to be more reliable in all cases. This leads us to believe that the original dataset presents mislabeled observations, which is concerning, as such instances are most likely used in the training process as well. Furthermore, we can observe that the same face appears twice in the test set, implying that the original dataset also contains duplicates.

Figure 5: Misclassifications with difference between predicted age and real age higher or equal to 40 years (MobileNet)

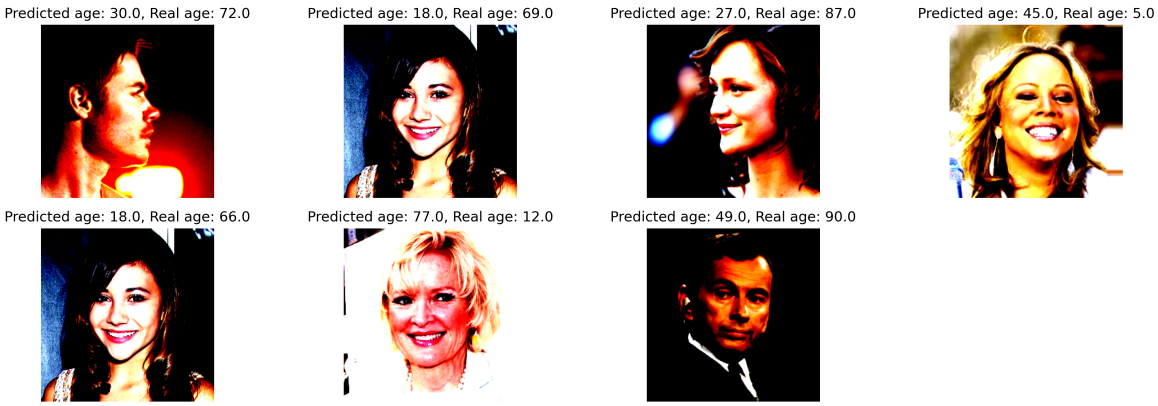

Figure 6: Misclassifications with difference between predicted age and real age higher or equal to 40 years (EfficientNet)

Having assessed that the biggest errors are due to issues in the dataset, we can now visualize in Figure 7 and Figure 8 a random sample of eight faces misclassified within the 15 to 20-year age range. Overall, most of the misclassifications made by the networks in this age range seem to involve faces whose ages are challenging to predict even for a human observer.

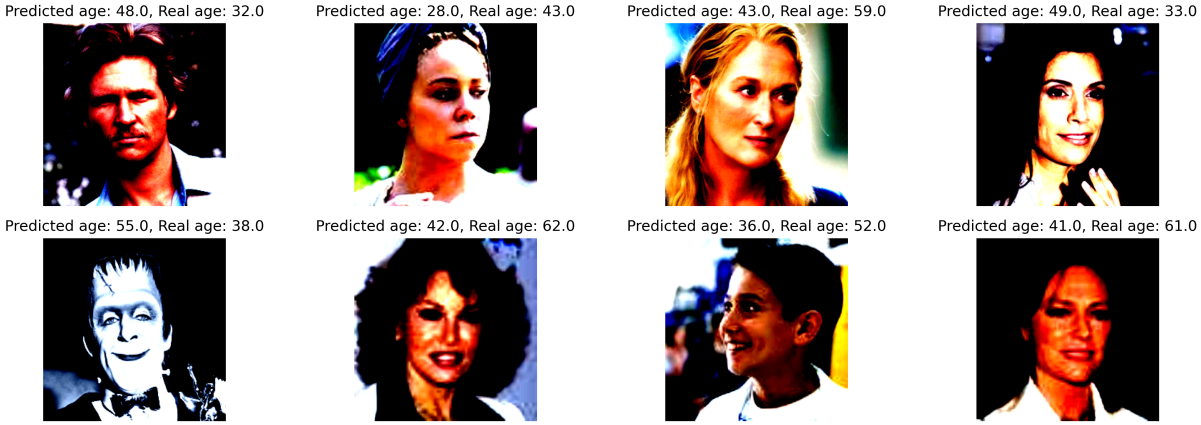

Figure 7: Incorrect classifications with a difference between predicted age and real age ranging from 15 to 20 years (MobileNet)

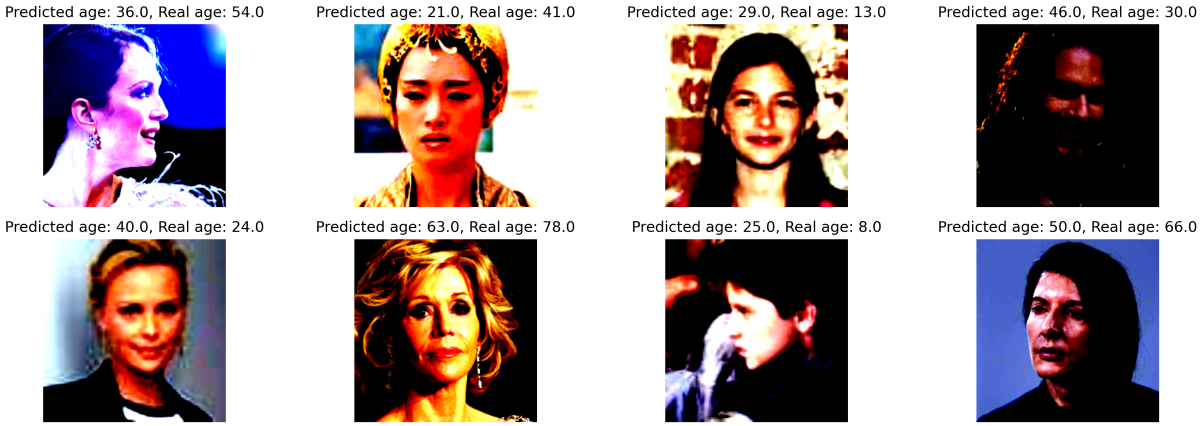

Figure 8: Incorrect classifications with a difference between predicted age and real age ranging from 15 to 20 years (EfficientNet)

5. Conclusions:

By working with the IMDB dataset, we were able to test the power and the efficiency of transfer learning: despite having been trained on a different dataset, the network was able to adapt to our data and achieve an impressive regression performance. We observed how, if major computational power is available, performing fine-tuning on all the layers of the network will surely lead to better results.

While investigating misclassifications, we identified two major dataset-related issues: mislabeled observations and duplicates. These issues are concerning as they introduce incorrect information into the training process, impacting the model performance. Detecting mislabeled data in such dataset poses a significant challenge, especially considering the dataset's scale. On the other hand, addressing the duplicate problem could be possible by implementing appropriate preprocessing steps.