COMPUTER VISION (2023-2024)

REPORT: **Task 1**

Group members:

Joel Dieguez (1), jdieguvi15@alumnes.ub.edu , joeldvd
Clàudia Valverde (2), clvalves7@alumnes.ub.edu , claudia4

1. MODEL DESCRIPTION

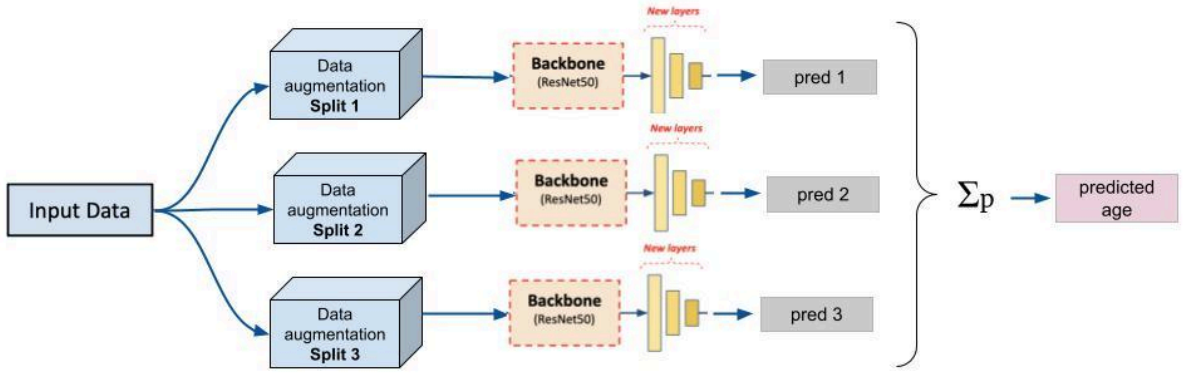

Figure 1: Illustration of the proposed model.

Our proposed model consists of an ensemble of 3 copies of the same model trained on different splits of the data, which received data augmentation to be balanced. The final prediction is the average of the 3 results. The best part of this model is that it allows us to work with a lot of data augmentation while not surpassing the RAM limit on Colab. Also, the ensemble tends to mitigate any occasional bias a model has picked up.

2. BIAS MITIGATION STRATEGY

Firstly, we have done an analysis of the data in order to understand what we are dealing with. It is important to check if the training and validation datasets follow more or less the same distribution, as well as check the distribution of the metadata. In order to mitigate bias, we have decided to perform data augmentation based on the results we observed from data analysis.

We decided to apply 17 different transformations, listed in **Table 1**. Some of them are better not combined at the same time, for example zoom in and zoom out, so we grouped them and at most one of them is applied each time. To apply them, we created a function that goes through each group and rolls a random number to decide which if any is applied.

Our objective is to augment the data based on age, ethnicity and facial expression. For this objective, we defined a function that decided how many times each image should appear in the final dataset to make it balanced. The copies of the image generated using randomly the transformations above. The percentages are specifically thought for them to be meaningful and different but not lose the main characteristics.

This function iterated many times, from a naive implementation based on age to a full one which takes into account all metadata. However, the main problem was that the dataset was

too big for Colab and the RAM wouldn't allow loading it and the model at the same time. This is the reason we decided to split the data into three splits. All of them contained all the "vulnerable" images. We define a vulnerable image as one that belongs to a poorly represented dataset, like "afroamerican" or "older than 60 years old". The other images are split evenly between the three datasets. All of them are further represented using data augmentation. Therefore, all data is taken into account but each model only sees a balanced representation of the dataset.

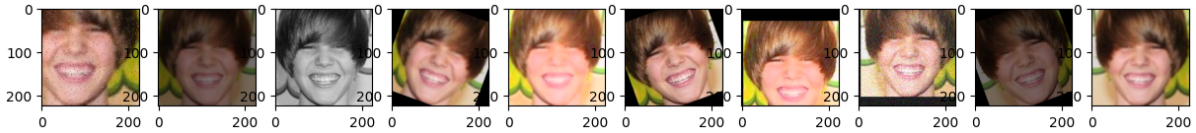

Figure 2: Example of the effects of the data augmentation on an image.

Type	Transformation	% activation
Flip	Flip	50%
Sharpness	Blur	33%
	Sharpen	33%
Color modification	Brighten	20%
	Darken	20%
	To grayscale	10%
	Reduce random color channel	4%
Linear transformations	Zoom in	10%
	Zoom out	10%
	Random zoom (in random quadrant of the image)	10%
	Rotate left	10%
	Rotate right	10%
	Random rotate (focusing on random quadrant and with random angle in a range)	10%
	Translocation	10%
Noise	Gaussian	10%
	Salt and pepper	10%
	Speckle	2%

Table 1: All the transformations applied, grouped by type of transformation and their chance to be applied in each group, if none was selected, the image remained equal.

3. TRAINING STRATEGY

Inspired by the starting kit, we adopted a training strategy divided in three stages. Each data split follows the same training procedure but using different splits of data at each stage.

1. During the initial stage, we leverage the pretrained ResNet50 model by freezing the already trained layers and training the newly added ones.
2. The second stage involves training the entire model concurrently on the augmented dataset.
3. Lastly, the third stage encompasses training the entire model on the original dataset for only 10 epochs with the idea for the model to remember the original data.

After this, we compute the average probability of the 3 splits in order to obtain a predicted age.

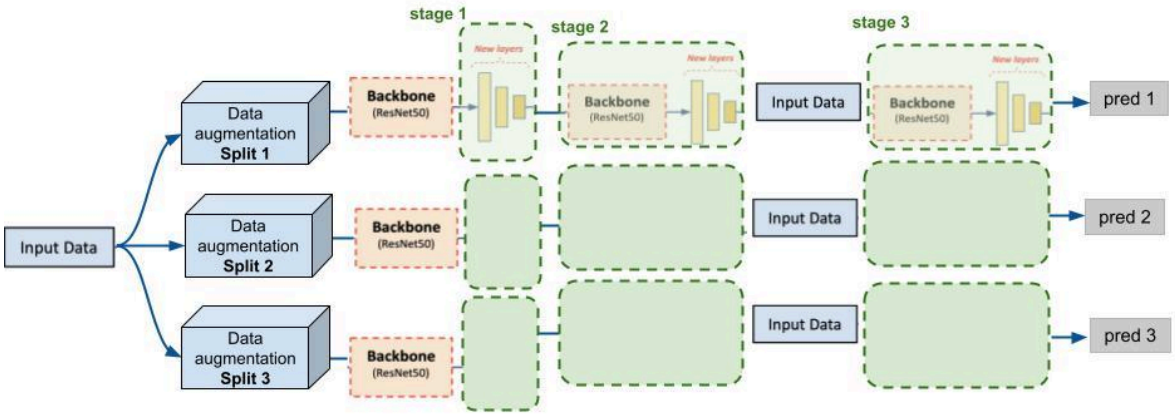

Figure 3: Training strategy of the selected model.

4. EXPERIMENTS AND RESULTS

EXPERIMENT 1: Data augmentation in general with starting-kit model.
This experiment consisted in ensuring that the data augmentation implemented was working properly, following the procedure from the starting-kit, we trained the model using our soft and extreme Data Augmentation.

By following this procedure, we achieved a higher overall accuracy, however, the bias did not mitigate. In the beginning, our transformations were very extreme and with high probability of occuring, resulting with many images very far away from the objective dataset. Even though, with this metric, we got better MAE. We also considered another set of transformations, more subtle and with smaller activation rates. Mind you that the transformations themselves and the activation rates in **Table 1** are more exaggerated in the extreme case, but we are not going to go into detail here, please see it in the code.

EXPERIMENT 2: Fine-Tuning based on age groups with data augmentation for each subgroup.
With the conclusions extracted from the data analysis for this experiment we decided to focus on each of the age groups, fine-tuning Stage 1 model four times, as shown in **Figure 4**. The goal is to develop four specialized models tailored to distinct age ranges. Our data analysis revealed significant variations in metadata distribution according to age groups, prompting us to apply data augmentation specific to each age range.

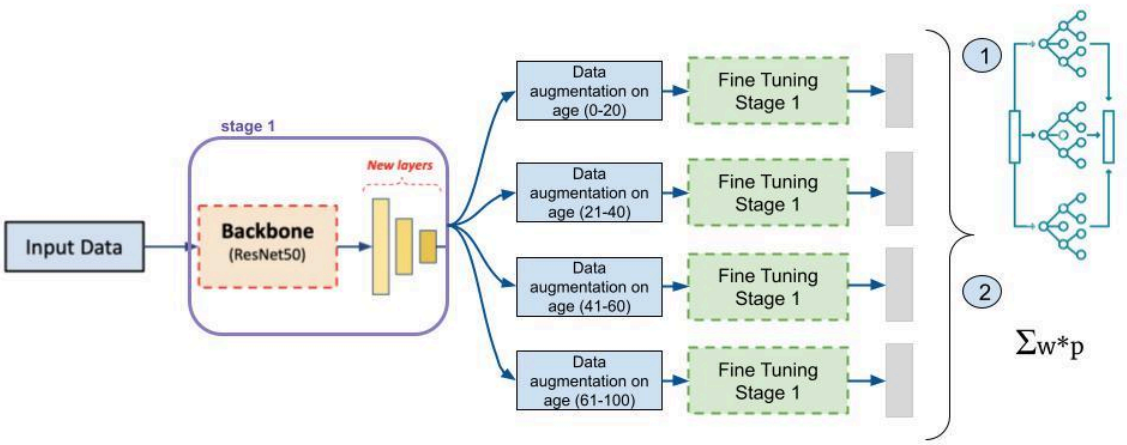

Figure 4: Diagram architecture and training routine of the experiment 2.

To merge the models, we tested 2 different procedures: 1) stacking, consisted on stacking the validation predictions of each model and training a RandomForestRegressor using these stacked predictions. 2) We constructed a weighted average function to compute the average age of each sample, considering the age range it is predicted to fall into. In this way, predictions from models covering the specific age range of a sample were given more weight when computing the average.

Although the accuracy for each individual model in its age range was good, the merging of the models struggled, as shown in **Table 2**.

EXPERIMENT 3: Data splitting + 3 stage training
This is the experiment that we have selected as the best model and training strategy, it has been already explained in previous sections. In **Table 2** we can observe a comparison on CodaLab test data of all experiments.

Experiment	Version	Data Aug	Gender (bias)	Expression (bias)	Ethnicity (bias)	Age (bias)	Avg bias	Test MAE
	Baseline	No	0.154	**0.1023**	**0.3441**	3.1021	**0.9256**	4.8119
1	extreme	Yes	**0.0401**	0.343203	1.1946	**2.4392**	1.0042	4.5194
1	soft	Yes	0.3210	0.106047	1.5810	2.6335	1.1603	**4.3719**
2	w/ stacking	Yes	3.9887	2.436160	4.4954	33.351	11.0678	32.729
2	w/ weighted prob	Yes	4.0101	2.451201	4.5123	33.481	11.1136	32.770
3	split mean	Yes	**0.0401**	0.396389	0.6124	3.8192	1.2170	5.1558

Table 2: Summary of test metrics for all experiments.

Experiment	Version	Data Aug	Gender (bias)	Expression (bias)	Ethnicity (bias)	Age (bias)	Avg bias	Test MAE
	Baseline	No	0.154	0.1023	0.3441	3.1021	0.9256	**4.8119**
3	split 1	Yes	0.2212	0.388112	0.6038	2.4562	0.9173	6.2942
3	split 2	Yes	0.5853	**0.062292**	**0.1995**	**1.6468**	**0.6234**	5.2297
3	split 3	Yes	0.2876	0.339618	1.0263	3.0063	1.1649	5.4549
3	split mean	Yes	**0.0401**	0.396389	0.6124	3.8192	1.2170	5.1558

5. FINAL REMARKS

Through our experimentation and analysis, we encountered challenges in improving the starting-kit model and addressing biases effectively. Despite exploring a wide array of strategies, including various model combinations, training routines, and data augmentation techniques, we were unable to achieve significant advancements. We explored more simple augmentation procedures which could lead to cleaner data and potentially better model performance, and more extreme ones, which introduced more variations. Regarding training strategies, we realize that simpler approaches may be more effective. Despite experimenting with complex training strategies, we did not observe notable improvements over the starting-kit model. However, we believe that EXPERIMENT 2 and 3 strategies hold promise.

Nevertheless, overfitting on augmented data proved to be a significant challenge. Future efforts could focus on exploring regularization techniques to mitigate this issue.

Our biggest challenges were due to the use of Colab. Issues such as RAM constraints and lack of access to GPU instances restricted our ability to conduct experiments effectively.

Our future work will involve exploring different training procedures and fine-tuning hyperparameters to identify optimal configurations. Additionally, we have a lot of faith in the model proposed in EXPERIMENT 2, with another merging strategy and more testing. Furthermore, we also want to evaluate new model architectures such as Xception as well as try other pre-trained image datasets such as Face instead of ImageNet.