NOVA IMS 2021/2022

DEEP LEARNING PROJECT

Group 8
Bruna Duarte, 20210669
Francisco Ornelas, 20210660
Isha Pandya, 20210920
Lucas Corrêa, 20211006

Task definition

The goal of this project is to build a Convolutional Neural Network that predicts the age and gender of a person by processing a picture of their face. There are plenty of real-word applications for this problem. For instance, it could help the police force identify suspects by these attributes or customize a service according to the person's age and gender.

In essence, our model will receive a .jpeg image of the front of a person's face and try to classify first as male, female or neutral (for babies that don't show very strong traces of any gender) and then in which bins of age this person is most probably is: 0-2, 3-6, 8-13, 15-24, 25-34, 35-43, 45-100. The bins of age are separated according to the available data that is taken from the OUI-Audience Face Image Project[1] which compiled Flickr photos albuns uploaded from smartphones devices of people that allowed them to let their images to the general public.

Evaluation measure

For this project, we will mainly pursue higher accuracy rates as the main method of validation. Nevertheless, we also compare the loss, f1 score, overfitting and computing time between different models' architecture in order to provide a full view of how each version of the model improved.

Since this experiment already has a pre-processed database of images with the related labels, our group used the results of different papers with the similar task that used the same database in order to benchmark our work improvement. The related papers are in the last section of this report.

Approach

The task was divided by creating two separated models instead of a single one like *Gil Levi and Tal Hassner, 2015.* With this decision, we expected each convolutional network to incorporate the specifics attributes for each type of classification task.

Before creating the model we combined the information of image id and label that are in the source as a txt file. In the combined dataframe we dropped the images ids without labels and re-assigned age labels outside bins to the correct bin (eg.: label with only 34 was assigned to the bin 25-34). Once we had a clear dataset with the images ids and labels we preprocessed the images into specific folders according to the stratified split of the data.

After the initial processing of images and labels, we develop a process to systematically compute and save the results for each model. This process involved the use of checkpoints of each epoch through callbacks and a function that would produce the measures discussed in the last section.

Following the standard approach for a problem like this one, we started out with small Convolutional Neural Networks and progressively increased its size and depth. As always, this process involved a lot of trial and error and some arbitrariness in our choices. Nonetheless, we tried to follow some general guidelines such as keeping the first layers with less filters and progressively increasing the number of filters in the following layers. This, in view of the fact that, with a few small recognized features (ex: lines, etc.), you can create a lot of different relevant shapes, hence the increasing number of filters in the last layers, as well as their increasing size.

[1] https://talhassner.github.io/home/projects/Adience/Adience-data.html

Gender Models

The first few models we tried ended up performing well in the **gender** prediction task (maximum of 77,74% of validation accuracy). This was a small network with three Convolutional layers, three MaxPooling layers and 3 dense layers (for more detail please refer to the code). We wanted to see how far we could take this result and so started adding more layers, trying various different combinations of architectures, filters, and hyperparameters. It would not be feasible to post the stats and results of every single model we tried, so we decided to choose 3 for both of the prediction tasks, representative of three different steps in our development process.

We did notice that deeper networks, although more time consuming, didn´t necessarily equate to better results. Model gender 2 for example, is a lot deeper than model gender 1, but performed significantly worse and was more time-consuming while also having a more drastic overfit (Table 1), which might be the result of more parameters getting too specific on the training set and not generalizing well enough.

To try and address this situation, we implemented various different techniques in these bigger models to try to reduce overfitting. These were L2 kernel regularizers, which add a small penalty to the weights which in theory attempts to generalize better, and Dropout Layers which artificially nullify a given percentage of layers again to reduce overfitting. Another addition were the Batch Normalization layers, which in theory, help against a phenomenon called internal covariate shift, which cause, among other problems, increased training time by scaling the output of the layer. These techniques proved useful and after determining the specific hyperparameters, the chosen model(gender 3) was in fact the best performing with a maximum validation accuracy of 83%(Table 1) and **visually** the one with less overfitting.

With regards to the trade-off between accuracy and efficiency, it will really depend on the specifications of the user´s problem. The model gender 2 is definitely not usable in this context under any circumstances, given that the gender 1 model is around 2h30m faster to train and still has better accuracy. It is also feasible to think that for some tasks, the difference in accuracy between the gender 1 and gender 3 models (around 8% - Table 1) is not enough to account for the difference in training time (25 min compared to 7 hours - Table 1).

Age Models

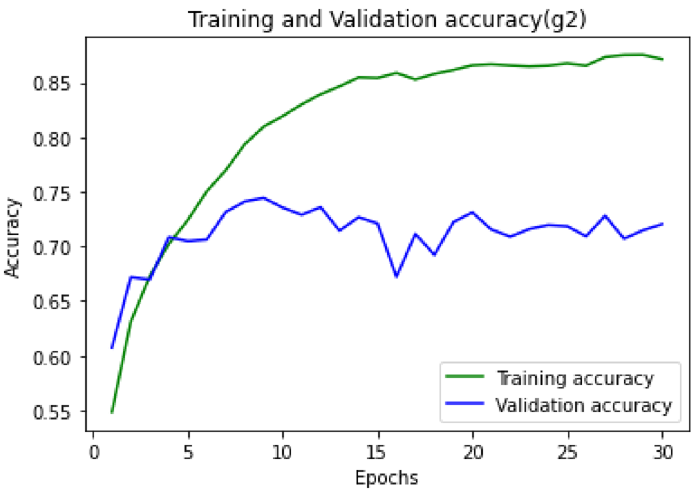
As far as the age models go, the situation is a bit different. The first family of models, represented by the Age 1 model) with a comparatively simple architecture did not perform as well as its more complex cousins(as seen in Table 1, had a validation accuracy of 41%). Perhaps because this was a more demanding task, with 7 different output categories, so more depth was needed. As such, we followed the same logic as in the gender task, and tried to increase depth and size until figuring out the best parameters given our evaluation metric of accuracy. This process led us to the model Age 2, which outperformed the Age 1 model by almost 20% in terms of accuracy albeit with an increase of 4h30 hours in training time.
When we tried to add some techniques to counter over-fitting (in this particular case 2 Dropout Layers), creating the Age 3 model, the accuracy decreased, and the training time increased, perhaps indicating the Age 2 model was not overfit enough to benefit from this strategy, or that the neurons being nullified by the Dropout layer were significantly helpful.

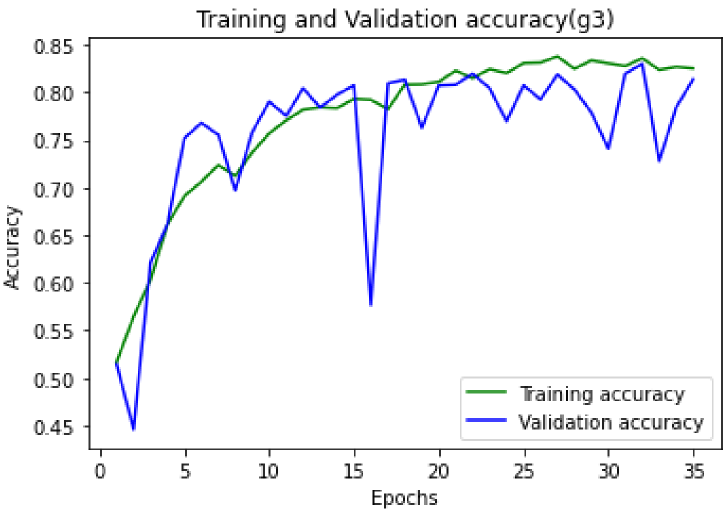
Error analysis

In order to compare and calculate our chosen measures, we split the dataset into train, validation and test images, with 70%, 15% and 15% of the data respectively. With this split of the data we hoped to (1) evaluate the overfitting, (2) improvement between models and (3) capacity of the model to generalize the task for other images.

The overfitting was analyzed comparing the validation and train accuracy in each epoch by a graphic approach. For instance, as we got a better accuracy rate in the second versions of the models, we started to see a higher gap between those two metrics, as seen in the graphic below for the second version of the gender model.

In this way, the improvement for the third version of the models was to approximate these two measures and decrease the overfitting, which was done as shown in the left graph below.





Even though the validation accuracy is a good way to compare models and overfit, we separated a test set of images to test the predicted power of the models in a totally unseen set of data.

References

Gil Levi and Tal Hassner, Age and Gender Classification Using Convolutional Neural Networks, IEEE Workshop on Analysis and Modeling of Faces and Gestures (AMFG), at the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Boston, June 2015

Eran Eidinger, Roee Enbar, and Tal Hassner, Age and Gender Estimation of Unfiltered Faces, Transactions on Information Forensics and Security (IEEE-TIFS), special issue on Facial Biometrics in the Wild, Volume 9, Issue 12, pages 2170 - 2179, Dec. 2014

T. Hassner, S. Harel, E. Paz, and R. Enbar. Effective face frontalization in unconstrained images. Proc. Conf. Comput. Vision Pattern Recognition, 2015.

Annex

Table 1

Task Model / Version	Version 1: Basic Model	Version 2: Increased Layers	Version 3: Increased layers reducing overfit
Age	Metrics: train_acc: 37% val_acc: 41% f1:0.22 Arc.: 3 conv. layers 3 dense layers all pool size of (3,3) ~460.000 params. ~30 min running time	Metrics: train_acc: 67% acc_val: 59% f1: 0.35 Arc.: 5 conv. layers 5 dense layers 3 pool size of (2,2), 1 of (1,1) and 1 of (3,3) ~1.6MM params. ~5h running time	Metrics: train_acc: 52% acc_val: 53% f1: 0.40 Arc.: 5 conv. layers 5 dense layers 4 pool size of (~1,9MM params. ~7h running time
Gender	Metrics: train_acc: 84%, val_acc: 75%, f1: 0.44 Arc.: 3 conv. layers 4 dense layers pool size of (3,3) ~49.000 params. ~25 min running time	Metrics: train_acc: 86%, val_acc: 72%, f1: 0.47 Arc.: 5 conv. layers 5 dense layers pool size (2,2) 3,5MM params. 3h min running time	Metrics: train_acc: 84%, val_acc: 83%, f1: 0.55 Arc.: 3 conv. layers 4 dense layers pool size of (2,2) 3,6MM params. 7h running time