CSC413 Project Proposal: Age and Sex Detection

Yongzhuo Xie; Shimiao Wang; Jihong Huang

March 2022

1 Abstract

Due to the extremely increasing of social media and internet, the automatic age and gender classification has become very relevant to a huge of applications. So in our paper, we will train and test two different CNN algorithms to check how they work in different situations and compares the performances. Two algorithms are based on $LeNet-5$ and $VGG-16$ respectively. The first one is very simple to implement while the other one used some transfer learning.

2 Introduction

Age and sex prediction has been a hot topic in the field of image recognition thanks to the introduction of deep learning techniques and the rapidly increasing amount of images uploaded on the Internet. This domain of recognizing attributes of age and sex has great potentials the in fields like Human Computer Interaction and IoT, which always desire better facial recognition for better user experience and security checks.

However, this task is far more difficult than one may imagine. Although humans are good at determining obvious and dichotomous features like sex, age estimation remains formidable for us. Such difficulty mostly comes from the fact that aging of face is determined by various factors including genes, lifestyles and environment. As a result, different people of similar ages can have distinct facial images, which makes age estimation a hard task even for an experienced expert.

Therefore, in this paper, we would explore different methods and algorithms to learn super-human capabilities in recognizing age and sex using deep learning, and then compare their effects. Moreover, certain extension will be applied on these algorithms to use them in a new domain.

3 Related Work

Early-staged works of age estimation often used anthropometric methods along with models like SVM, which focused on the explicit facial features including wrinkles and distance between eyes. Some examples are *PCA*, Blocking *ULBP*, and *LDP*. Such methods usually involve particular data augmentation on the input images to guarantee a generally meaningful measurement. Undeniably, these anthropometric methods are somewhat guaranteed by expert theories in human facial features. Their performances were satisfactory of the day. However, they are vulnerable to external factors like lighting, contrast, angle, and image resolution even with high-cost augmentation. Besides, the generalization is not very guaranteed due to the varying features of different people of similar ages.

Recent introduction of convolutional neural network to this field of age prediction makes great performances. One of the first few applications was $LeNet-5$ network, which is famous as the pioneer of CNN. Later, the paper introducing transfer learning using pretained $VGG-16$ made excellent generalizations, which put the deep learning increasingly applied in the age estimation. Indeed, deep learning like $VGG-16$ could provide more robust feature extraction which then give better generalization. And one of our main goals is to implement and compare the performances of $LeNet-5$ and $VGG-16$ in our task domain.

In comparison, sex prediction is a easier task for us to implement and algorithms like SVM would be enough for the purpose. But more robust features from CNN can always give better generalizations on sex prediction especially in those extreme cases where facial features are sexually neutral.

4 Methods and Algorithm

Our first algorithm used is a kind of CNN based on $LeNet-5$. It is simple and we will use it for both sex and age detection. Generally, this network consists of only three different convolutional layers and two fully-connected layers with some max pooling layers and ReLU functions.

The first convolutional layer is defined as 96 filters of size 3*7*7 pixels followed by a linear operator (ReLu), a max pooling layer taking the maximal value of 3*3 regions with two-pixels strides and a local response normalization layer. The second convolutional layer contains 256 filters of size 96*5*5 pixels. Again, followed by a ReLU, a max pooling layer and a normalization layer as before. The last convolutional layer applies a set of 384 filters of size 256*3*3 pixels and followed by ReLU and a max pooling layer. The first fully-connected layer receives the output of the third convolutional layer and contains 512 neurons, followed by a ReLU and a dropout layer. The second fully-connected layer receives the 512 dimensional output of the first fully-connected layer and contains 512 neurons, followed by a ReLU and a dropout layer again. The last fully-connected layer maps to the final classes which represents age or gender. Finally, the output from last fully-connected layer is transferred to a soft-max layer that calculates the probability for each class.

The second algorithm used is based on Transfer learning. We chosse UTKFace as the evaluation dataset. We use convolutional blocks of VGG16 pretrained on VGGFace dataset as feature extractor. The model is originally used for facial recognition, thus it can be used for feature extraction in this case. The network VGG16 originally has a feature extraction layer (comprises 30 layers) and a classifier layer (comprises 6 layers) where the classifier layer will be substitute by our own layer in this case. Following will be the network architecture be used in the model to train on top of features extracted. For task of gender classification, the network architecture comprises of 3 fully connected layers, each containing layers has spatial drop out (the purpose of dropout is to decrease the over-fitting and enhance the overall performance of the neural network) with probability of 0.2, 0.2 and 0.1 with ReLu activation then finally the output layer has 2 neurons with soft-max activation function. We use cross entropy as the criterion of the loss function for gender classification. Fro the task of age estimation, Moreover, there is a base line model: a custom CNN we construct similar to the structure of VGG16 but fewer layer. We also perform some evaluation on this simpler model in comparison with the previous model with VGG16.

5 Experiments and Discussion

5.1 CNN

The original dataset used in the paper included 26,580 in-wild images of 2284 subjects, where each image has its label of corresponding age group out of (0-2, 4-6, 8-13, 15-20, 25-32, 38-43, 48-53, 60-100) and gender. After resizing the cropped images and removing the images without clear labels, we build a training dataset of 12,983 training images and 3,375 testing images where each image size is $227 \times 227 \times 3$. Considering the actual computational cost, in our experiments we applied 5,000 images for training and 1,000 for testing.
Our training used CrossEntropyLoss as loss function, Adam optimizer with learning rate 0.0005, 2 epochs and batch_size=50. During the training process, the best training accuracy is 0.38. The test accuracy is 0.332. Given the limited resources, we are unable to tune the number of epochs as we usually do. However, in the countable trials, it showed that there is no meaningful changes after 3 epochs. During the tuning of the learning rates, a larger learning rate could result in unstable performances like divergence or oscillations. Instead, a lower learning rate like 1e-5 makes the model not learn anything in the training and finally give a test accuracy around 0.2, which would be close to random guess (0.125). Besides, the changes of batch_size seem not to influence the model much, which is a result of our choice of low epoch. If we choose a large batch size over 500, there would be only several updates on the model parameters. And the batch sizes below 200 showed similar performances.
For the extension of our model, we changed the output layer of our model and then applied on the gender estimation of the same training and testing images. The test accuracy this time is over 0.7, where the improvement might come from the fact that there are only 2 classes here to be predicted.

5.2 Transfer learning with VGG16

The originally image from UTKFace dataset has a a shape 200x200x3, and the dataset has a more than 20,000 images. We notice that this dataset is very computationally costly during the training, so we find a compressed version of the UTKFace dataset with each image only has 48x48x1 large as our evaluation dataset.

5.2.1 Custom CNN

As mentioned in part 4, we created the simple custom CNN model without transfer learning first.
Training and Testing:
For age estimation, the loss function we used is mean-squared error (MSE) and cross entropy as the criterion for gender classification.
We have trained some models with different hyper-parameters, and we record the one with the best performance.
We used 32 as the batch size and 30 epochs in total to train the model. Also, we initialized the learning rate to 0.00001 and we used learning rate decay method during the training to make sure that as training proceed, the learning rate could be still small enough to reach the local minimum. The learning rate changes to 0.6 times the previous rate per 5 epochs.
Below are the two graphs that recorded the lowest loss against epoch with the hyper-parameters described above:
We can clearly see that the loss goes down as epoch increases.
The model has the final loss on age estimation test set is 35.6, and the model has the final accuracy on gender classification is 0.7.

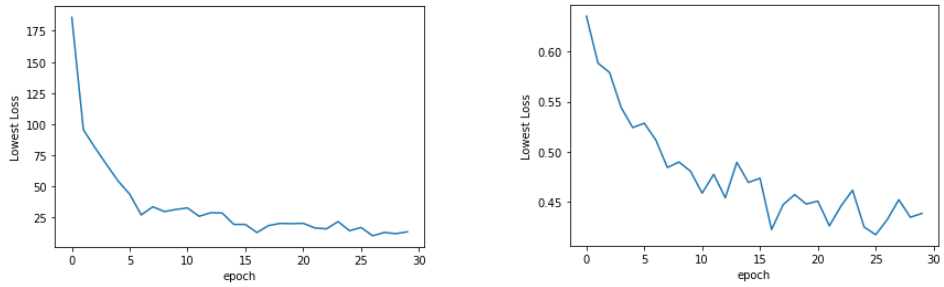

Figure 1: Training plots depicting loss by the epoch of (Left) Age Estimation (Right) gender Classification

5.2.2 Transfer Learning

For the task of gender classification, our loss criterion will be cross entropy .We find that the best hyper-parameters during the training process is epoch=10, batch size = 128 and learning rate set to 0.0001. As you can see the training loss gradually decrease during the training process but it starts overfitting after the 8 epoch, so we stop at there.

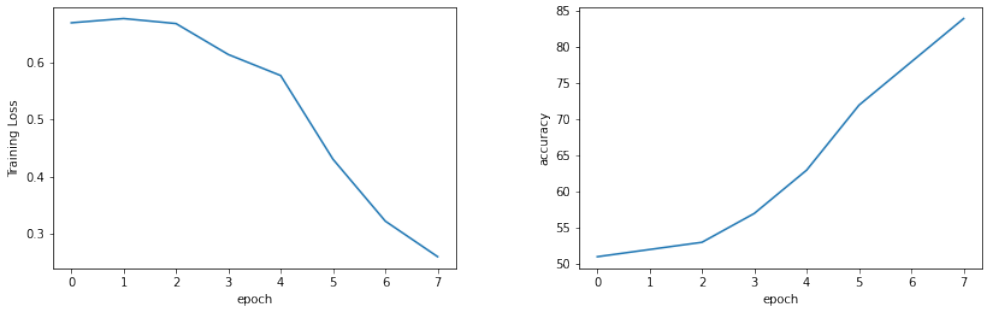

Figure 2: Train loss (left) Validation accuracy (right) both with respect to the number of epoch

More discussion about the freezing pre-trained layer: We have try to train the model if we freeze the layers of from VGG16 during the training process. It only achieve around 52% accuracy on the test set. It spends less computation time during the training process but the result is not quite well. So we believe freezing the pre-trained layers does not help with the model performance.

6 Conclusion

In this paper, we proposed two ways to deal the problem with age estimation and gender classification, a simple CNN based on $LeNet-5$ and transfer learning using $VGG-16$. From a reproductive perspective, our model gives an accuracy around 10 percent lower than that of the model in paper in both domains. Although such a result is not satisfying, but I think it is reasonable given the difference on computation resources. This drop in the performance may also support that this naive model is too general to work on a complex domain like age estimation by directly considering all the pixels in an image.

7 Contributions

Yongzhuo Xie: Mainly response for the coding part of the transfer learning and the written part of the transfer learning.
Shimiao Wang: Written the part Abstract and reference; introduce the Methods and Algorithm for our first CNN algorithm; and finish the experiment 5.2.1 Custom CNN with some results.
Jihong Huang: Written the part Introduction and Related Work; rewrite the simple CNN in the first reference by Pytorch (originally made by Caffe) and conducted the experiments in 5.1.

8 Reference

[1] Age and Gender Classification using Convolutional Neural Networks, Gil Levi and Tal Hassner (https://talhassner.github.io/home/projects/cnn_agegender/CVPR2015_CNN_AgeGenderEstimation.pdf).
[2] Age and Gender Prediction using Deep CNNs and Transfer Learning,Vikas Sheoran1,Shreyansh

Joshi2 and Tanisha R. Bhayani (https://arxiv.org/ftp/arxiv/papers/2110/2110.12633.pdf)

9 Appendix

Below is the Github repository for our project:
https://github.com/KolbeHuang/CSC413-Final-Project