Gender detection using Deep learning

Nahom Gebremeskel
Department of- Micro-data analysis
Dalarna university
Borlänge-Sweden
e-mail: h21nageb@du.se

Krishna Bharat Pasumarthy
Department of -Micro-data analysis
Dalarna university
Borlänge- Sweden
e-mail: h21pasbh@du.se

Abstract- Machine learning has a wide range of uses, including image recognition. It may be used to tackle issues in security, object detection, face detection, healthcare, entertainment, and other fields. This instrument has a great deal of potential to benefit our society, therefore it is critical to develop new applications for it, enhancing the existing techniques, and get more precise and practical insights from it. In this paper, we will look at and recommend the most effective method for determining gender using conventional neural networks (CNN). The networks are tuned with different parameters; however, Adam optimizer was the best one. In the experiment Convolutional and max pooling layers are combined and achieved 92.65 % classification accuracy.

Keywords; face detection, Inception v3, Convolutional neural network (CNN), Gender detection.

1. Introduction

In today's world, technological innovation has changed the dynamics of the way of living and how different skills are performed. With the steady growth of the population, image recognition has played its part to boost security systems in society. Object detection and recognition are among the tools that improve security. Facial images are quite helpful in acquiring information for different applications and human activities. Several researchers have been working and are still working to utilize this resiliency of human systems. According to Jin [1], different characteristics of a picture can be recognized by face biometric systems.

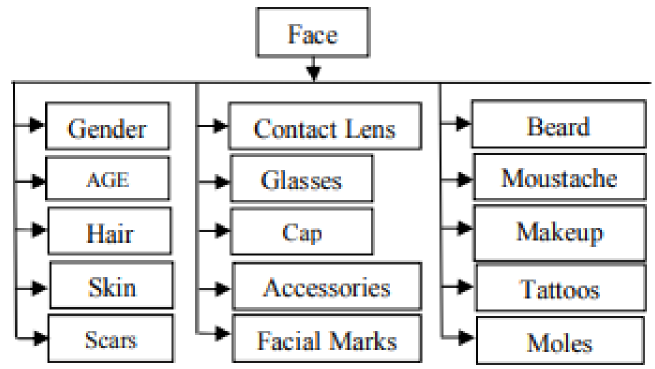

Fig-1: Auxiliary information from the face

The focus of this paper is gender detection using deep learning methods using a CelebA dataset. Studies have shown that deep learning tools performed well in recognizing the gender of individuals [1]. However, there are several difficulties for algorithms to predict gender accurately, to mention some facial expressions, background noise, etc. Therefore, there is a need to create dependable and accurate gender classifier algorithms.

Conventional systems of image detection consist of picture preprocessing, feature extraction, and more categorization. The performance of each system totally depends on the different parameters like the type of classifier, number of features, etc. The multilayer perceptrons of Conventional neural networks must have the algorithms to be more flexible by adding the accessibility to extract features and reduce dimensionality in the network [2].

According to Xie [2], based on the flexibility of CNN, accuracy has been improved for image recognition and analysis. In addition, training a CNN is easier than training classical classifiers, which has proved beneficial. Additionally, according to the authors, CNN uses fewer parameters than multilayer perceptron networks, which are resistant to shape and change [3]

In this paper, we will investigate and suggest the best and most efficient way of classifying gender using conventional neural networks (CNN). It has been shown that employing Adam optimization improves gender recognition from face patterns and allows training accuracy to converge in fewer epochs than other approaches currently in use.

The paper is classified into six sections. Section 1 describes the introduction to the paper, Section 2 gives a brief literature review of previous work, Section 3 explains the dataset information Section 4 explains the methodology and materials applied, Section 5 shares the results of the paper and finally, Section 6 provides the conclusion of the paper.

2. Literature review

Research conducted by Arigbabu [4] has implemented an analysis of gender recognition using facial images. A dataset known as Labeled faces in the wild (LFW) was utilized and achieved a 91% accuracy by utilizing a simple machine learning algorithm, Support vector Machine (SMV). Additionally, other authors [5] who have utilized the same dataset come up with 95 % accuracy after using a Fuzzy Linear discriminant classifier (LDA). The authors have used a Gabor filter which causes a higher accuracy in the gender classification task.

Research Levi and Tanner [6] have also implemented Conventional neural networks on a different dataset (audience benchmark genders) to train the algorithm to determine the age and gender. The model has achieved a mean accuracy of 83%. However, they have implemented different architectures and parameters to obtain high accuracy. But, when compared to the work of previously mentioned authors the accuracy of CNN is low.

Furthermore, an interesting publication that we cover was, an article written by Mirjalili and Ross [7]. The authors have conducted a gender and age detection on the Gender Feret dataset which results in 92 % accuracy using CNN. However, the dataset was easy to analyze and use. It is difficult to generalize the result of these experiments on more complex datasets.

The same idea that was applied by Mirijlili and Ross has been tested on a different dataset (ImageNet) by different authors [7]. Despite the Gender Feret dataset's great accuracy, it is apparent that ImageNet's performance has suffered. Additionally, overfitting is seen. Given that CNN employs a variety of parameters, it is important to always consider the possibility of overfitting.

Furthermore, authors Ozaki and Masaki [8] explained that the evaluation of Deep neural networks (DNN) improved the performance of models in recognizing images, objects, and several other tasks [8]. However, DNNs are affected significantly by hyperparameters. Therefore, slight changes in hyperparameters such as learning rate can differ the training results easily. Despite the complexity of the hyperparameters, DNN tuning by experts has demonstrated a significant improvement in the accuracy of DNN [8].

In this study, we solved the challenge of gender detection using deep convolutional networks, which aid in achieving high accuracy and yielding results that outperform several existing algorithms. Several hyperparameters will be considered as well.

3. Dataset

The dataset (CelebA) that was used in this study is publicly accessible at mmlab.ie.cuhk.edu [10]. The dataset consists of 84434 images of male and 118165 female celebrities. The photos were retrieved from the zip file. list_attr_celeba.txt is a text file that contains the 40 attribute details for each picture. Attribute names Male has values -1 and 1 to identify the gender and only this column is used from the txt file.

4. Methodology and Material

Data Handling: The original dataset is extracted after being obtained from the source. The dataset was also imported into Python for separating the required 20000 images. In this study, all attributes other than gender are ignored because gender identification is the main goal. The extracted images from the zip file have a default label as 0. So while finalizing the dataset for modeling, data labels are updated as 1 for male and 0 for female.

Exploratory Data Analysis (EDA): At the beginning of the experiment, sample images are previewed to help comprehend the data in depth and understand the details of the input channel. After performing exploratory data analysis, the dataset was found to be imbalanced and highly difficult to work with the entire dataset. As a result, we choose to balance the data by just photographing 20,000 images, 10,000 of which are male and 10,000 are female.

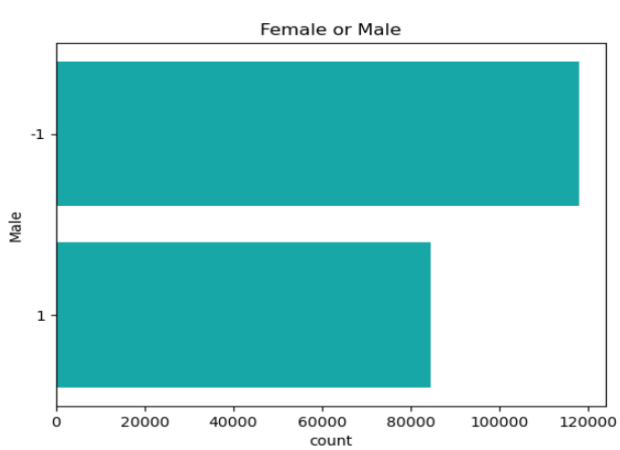

Fig-2: Data distribution.

Data split: The data is split in two 80% training and 20 % testing.

Data modeling: In the first stages of data modeling, we first set fundamental parameters with appropriate meaningful values for image size, batch size, number of workers for data loading, number of epochs, learning rate, momentum, number of needed output channels, and kernel size.

myCNN class was created with many layers as shown in fig 3.

```
----------------------------------------------------------------
        Layer (type)               Output Shape         Param #
================================================================
            Conv2d-1            [8, 5, 63, 63]              65
              ReLU-2            [8, 5, 63, 63]               0
         MaxPool2d-3            [8, 5, 31, 31]               0
            Linear-4                 [8, 2402]      11,544,012
              ReLU-5                 [8, 2402]               0
            Linear-6                    [8, 2]           4,806
================================================================
Total params: 11,548,883
Trainable params: 11,548,883
Non-trainable params: 0
----------------------------------------------------------------
Input size (MB): 0.38
Forward/backward pass size (MB): 3.01
Params size (MB): 44.06
Estimated Total Size (MB): 47.44
----------------------------------------------------------------
```

Fig-3: Summary of Model

ReLU is used as the activation function in the model developed with the parameters listed, since it often outperforms sigmoid and tanh functions. After the model was successfully created, a hyperparameter optimization has been tested to increase the model's efficiency. Further, Cross Entropy Loss function was implemented to determine the loss.

To improve the model and boost accuracy, the following hyperparameter tuning adjustments are made[11]:-

1. Batch Size = [8,16,32,64]
2. Kernel Size = [2,3,4,5]
3. Learning rate = [0.1,0.01,0.001,0.0001]
4. Momentum = [0.8,0.6,0.4,0.2]
5. Optimizer = SGD, Adam

6. Epochs = [2,3,5,10,20]

The value of each parameter is determined depending on the result of accuracy after tuning. For example, in this problem the best batch size was 32 with higher accuracy when compared with other batch sizes, therefore we tuned other parameters using the data loaded with that batch size. Further parameters are tuned using the model with updated parameters and so on.

5. Results

The initial model is created with batch size as 8, kernel size 2, learning rate 0.01, momentum 0.8, cross entropy loss function, SGD optimizer for 2 epochs. Figure 3 shows the first model information and the test accuracy found to be 50%. Hyperparameter tuning is performed to adjust the six different parameters and below figures shows the accuracies of each parameter for each set of values.

a. Batch size

The initial model was tested with different batch sizes as a result, it was observed that batch size 32 outperformed all other batch sizes with the result of 88.7 % accuracy.

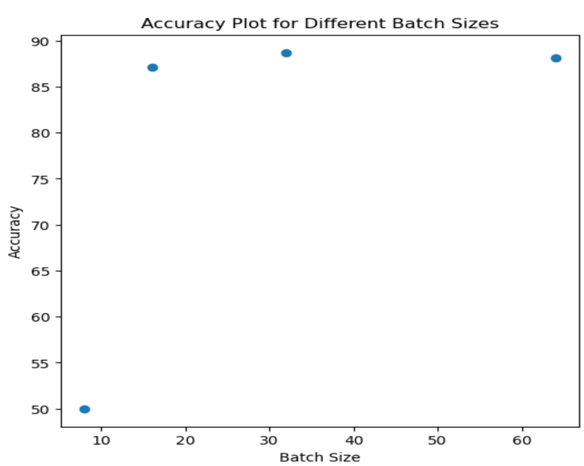

Fig- 4: Accuracy based on batch size.

b. Kernel size

Here the model with the best batch size is tested on different kernel sizes. It is noticed that the Kernel size 3 outperforms the other kernels taken in consideration with the accuracy of 90.97%.

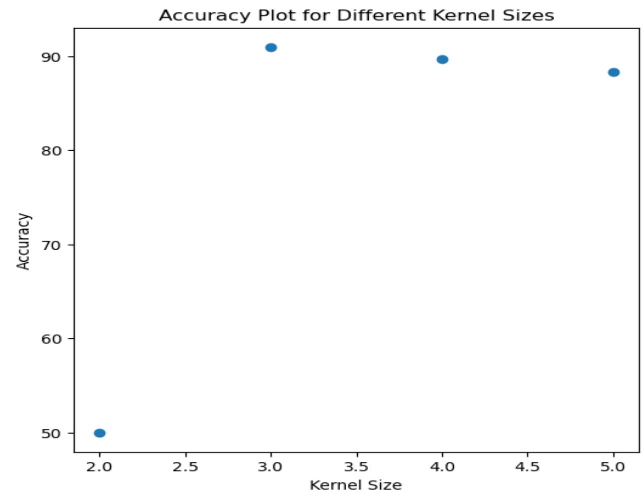

Fig-5: Accuracy based on kernel size

c. Learning rate

Learning rate plays a great role in the performance of a model. So in this study different learning rates have been taken into account and learning rate 0.01 scored the highest accuracy 90.8%.

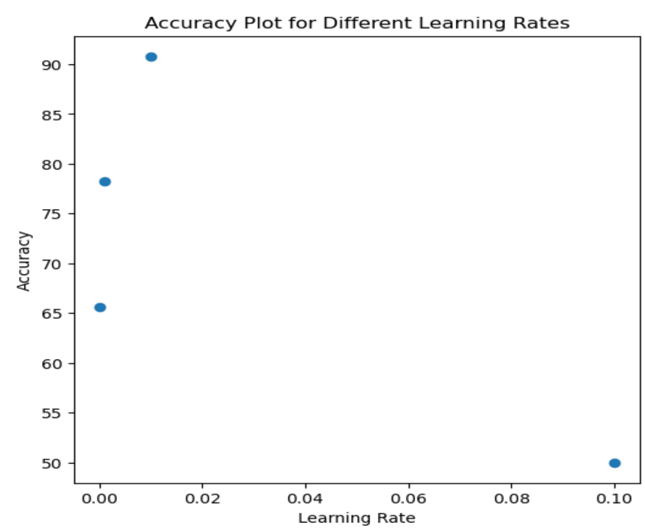

Fig-6: Accuracy based on learning rate.

d. Momentum

At this stage the model with the best result of batch size, kernel and learning rate is tested on different momentum. Momentum 0.6 scored 91.78 % accuracy over all the other models.

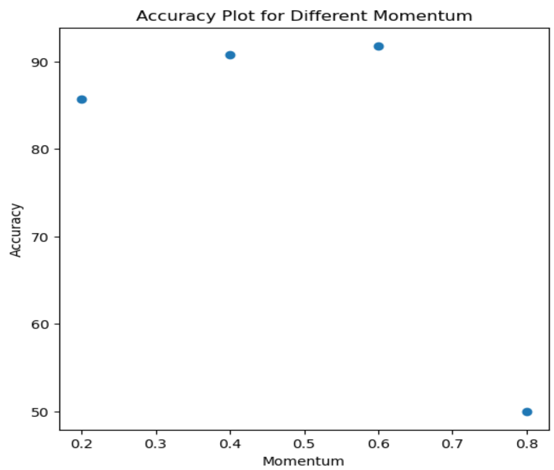

Fig-7: Accuracy based on Momentum

e. Optimizer

Furthermore, the previous best model with momentum also experimented on different optimizers and as result it is noticed Adam optimizer has outperformed Stochastic gradient descent with accuracy of 91.92%.

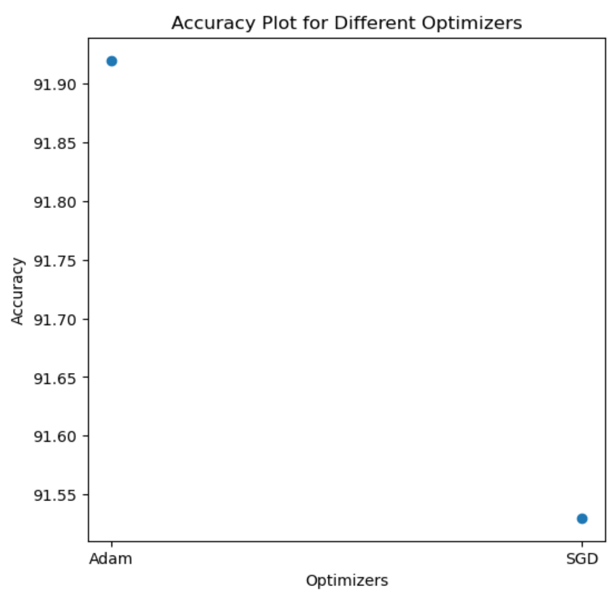

Fig-8: Accuracy based on Optimizer

f. Epochs

At last various epochs are also considered. It is observed that the model performance has improved greatly to 92.65% with 20 epochs.

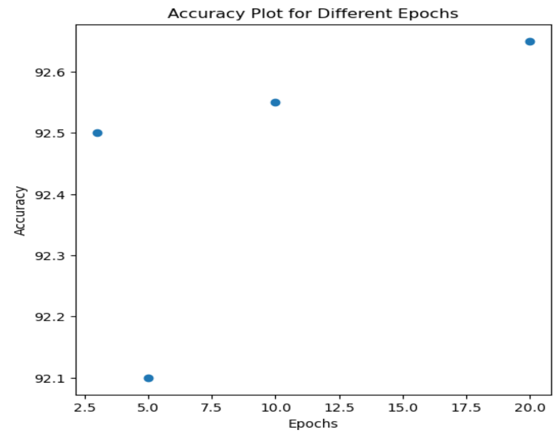

Fig-9: Accuracy based on Epochs.

Summary of result: So if we can summarize the result of the hyperparameter as below:

Hyperparameter Tuning			
Parameters	Values	Train Accuracy	Test Accuracy
Batch Size	8	49.52%	50%
	16	83.98%	87.08%
	32	86.79%	88.70%
	64	89.76%	88.15%
Kernel Size	2	49.52%	50%
	3	89.81%	90.97%
	4	90.06%	89.70%
	5	90.28%	88.35%
Learning Rate	0,1	49.52%	50%
	0,01	91.09%	90.80%
	0,001	75.54%	78.28%
	0,0001	63.13%	65.65%
Momentum	0,2	87.10%	85.57%
	0,4	89.02%	90.80%
	0,6	89.94%	91.78%
	0,8	49.52%	50%
Optimizer	Adam	92.35%	91.92%
	SGD	88.79%	91.53%
Epochs	3	91.99%	92.50%
	5	92.72%	92.10%
	10	94.73%	92.55%
	20	95.05%	92.65%

Table-1: Results of parameter tuning

6. Conclusion

Depending on the result of each parameter tuning that is presented on this paper it is evident enough to say models that take into account different parameters performed well. Tuning the parameters improved the accuracy from 50% to almost 93 percent. Finally, the results shown in table-1 highlights the best model for this dataset is the model which consists the following parameters:-

- Batch size= 32
- Kernel size= 3
- Learning rate= 0.01
- Momentum= 0.6
- Optimizer="Adam"
- Epoch= 20

Limitation

The authors of this paper are aware that not all parameter settings will always result in the greatest performance. This experiment has proved that using various parameter tuning techniques is helpful for attaining better performance generally. All parameters values used in this model can be used as default values for a similar dataset or this particular dataset. However, for different dataset it is necessary to test and compare various parameter values.

Reference

1. Jin, Jing, Bin Xu, Xiaoliang Liu, Yuanqing Wang, Liqun Cao, Lei Han, Biye Zhou, and Minggao Li. "A face detection and location method based on Feature Binding." Signal

2. H. Khalajzadeh, M. Mansouri and M. Teshnehlab, "Face Recognition Using Convolutional Neural Network and Simple Logistic Classifier", Advances in Intelligent Systems and Computing, vol. 223, pp. 197-207, 2013.

3. S. Xie and H. Hu, "Facial expression recognition with FRRCNN", Electronics Letters, vol. 53, no. 4, pp. 235-237, 2017

4. O. Ayodeji Arigbabu, S. Mumtazah Syed Ahmad, W. Azizun, W. Adnan, S. Yussof and S. Mahmood, "Soft Biometrics: Gender Recognition from Unconstrained Face Images using Local Feature Descriptor", Computer Vision and Pattern Recognition, pp. 1-9, 2017.

5. Chengjun Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," in IEEE Transactions on Image Processing, vol. 11, no. 4, pp. 467-476, Apr 2002

6. G. Levi and T. Hassncer, "Age and gender classification using convolutional neural networks," 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Boston, MA, pp. 34-42,2015.

7. A. Krizhevsky, I. Sutskever and G. Hinton, "ImageNet classification with deep convolutional neural networks", Communications of the ACM, vol. 60, no. 6, pp. 84-90, 2017.

8. Yoshihiko Ozaki, Masaki Yano and Masaki Onishi, "Effective hyperparameter optimization using Nelder-Mead method in deep learning"-2017

9. Charu C.Aggarwal "Neural Networks and Deep Learning"-2018

10. http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

11. https://towardsdatascience.com/simple-guide-to-hyperparameter-tuning-in-neural-networks-3fe03dad8594