Predicting Age, Gender, and Ethnicity Using CNNs

George Lu
Department of Computer Science
University of Central Florida
Orlando, Florida, USA
georgelu@knights.ucf.edu

Sarah Wilson
Department of Computer Science
University of Central Florida
Orlando, Florida, USA
sarahwilson@knights.ucf.edu

Abstract

In this paper we propose using an end to end Convolutional Neural Network to classify and predict age, gender, and ethnicity. We used a dataset based on the UTKFace dataset with over 20,000 samples to provide a large range of training and test data. We achieved relatively high accuracies for the gender and ethnicity predictions. The age prediction accuracy is lower than the gender and ethnicity models since it is hard to predict an exact age. However, the loss for this age model was relatively low. Overall, CNNs can make good predictions for these three categories.

I Introduction

A. Applications

As of 2019, there were 75 out of 176 countries using artificial intelligence in their surveillance operations [2]. These countries are using artificial intelligence to solve real-world issues such as video surveillance, security, forensics, and human-computer interaction. Especially important features, such as age and gender, are crucial in these real world examples. Over the past few years, many different methods have been proposed to solve these issues; some of them involve handcrafting features to solve the age, gender, and ethnicity classification problems.

However, these methods cannot handle the variations of each image along with the thousands of images for each specific case. That is why we propose using a CNN (Convolution Neural Network) to solve this problem because CNNs are able to handle wide variations in a dataset. This is why we propose using end-to-end deep learning models that predict age, gender, and ethnicity.

B. CNN Definition

A CNN is a class of neural network that have superior performance with images. They have three main types of layers: convolutional layers, pooling layers, and fully connected layers. [3].

A convolutions layer is where a kernel of size nxm is moved over the image like a sliding window. There are multiple kernels, and each of them will learn a specific feature. These convolution layers are really efficient because they exploit the fact that images have spatial data. This reduces the number of parameters it has to train, compared to a normal neural network that doesn't exploit spatial data.

A pooling layer is similar to down sampling in which it conducts dimensional reduction. This reduces the number of parameters in the next layer. We might lose some pixels but we try to keep the same amount of information by either performing a max or average pooling to extract the features while reducing the search space.

A fully connected layer is like a traditional neural network in which we just flatten the features and push it through a normal feed forward neural network. At the end, we either perform a soft-max activation for multi-class classification and compute the categorical cross-entropy loss for classification. Alternatively, we can compute the value for regression and use mean square error loss.

C. Challenges

One of the challenges we faced was that the image resolutions were small since they are 48x48. This made it hard to extract features. The model would probably have higher accuracy if the image resolutions were larger. We also couldn't apply a pre-train CNN because if we did, there would be too many convolution and pooling layers that the image wouldn't have any pixels left.

Another issue we dealt with was that since the dataset was unbalanced, it made it harder to predict the classes that have fewer samples. Since the White ethnicity samples were the majority, and the most populous age band was 24 to 29, the CNN would most likely be skewed towards predicting those categories. We also couldn't downsample, or else the dataset would have too few images. We couldn't upsample the other classes or else it would overfit the features in the over-sampled pictures.

We found that our CNNs were able to predict gender with a high accuracy of 91%. The Ethnicity prediction had a good accuracy of 80%, while the age prediction had an accuracy of 17% (with a ±1 age).

D. Related Work

Others have solved this problem using Convolutional Neural Networks. Levi and Hassner created a CNN that predicts the age and gender of a person in an image [4]. They decided to predict age based on ranges, not on a specific age. They were able to achieve an accuracy of around 78% for gender and 80% for age.

II Method

A. Dataset

The dataset used to train the CNNs [1] is based on the UTKFace dataset [5] by Yang Song and Zhifei Zhang. It has over 20,000 images with labels for the person's age, gender and ethnicity. Each image contains one person, who is positioned in the center of the image which has a height and width of 48 pixels. The labels for gender are male and female; the ethnicity labels are White, Black, Asian, Indian, and Other (Hispanic, Middle Eastern, Latino). The age range of the sample is 1-116 years.

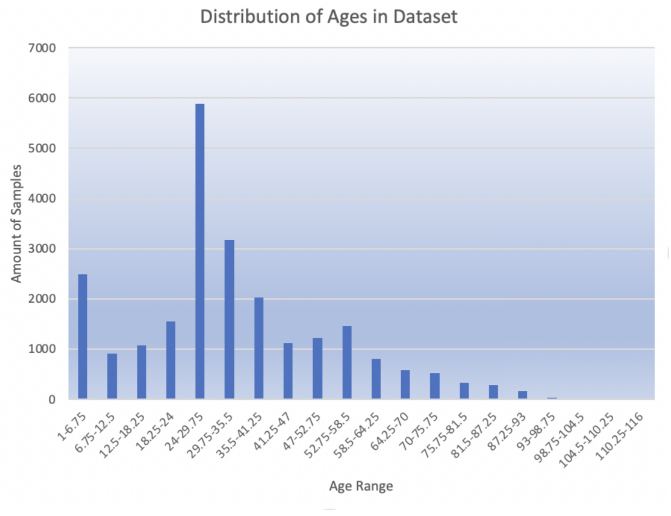

Fig. 1. Distribution of ages in the dataset

The dataset [1] is not comprised of equal samples for each category for age, gender, and ethnicity. The predominant age range is 24-29, which makes up 21.5% of the dataset shown in Figure 1. Meanwhile, the older ages of 93-116 make up a combined total of .24% of the dataset. For ethnicity, White is the majority class which is 36.9% of the samples. The second most populous class is Black which makes up 19.1% of the dataset as seen in Figure 2.

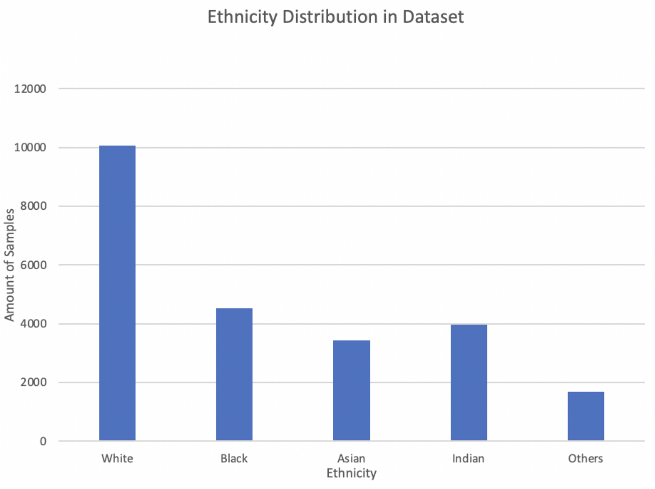

Fig. 2. Distribution of ethnicities in the dataset

Regarding gender, the split is 52.3% male and 47.7% female in Figure 3. Since the ages, genders, and ethnicities of the samples are not an even split, this might affect the CNN's output. Further testing would have to be done to show to what extent the classes with less samples are predicted with less accuracy.

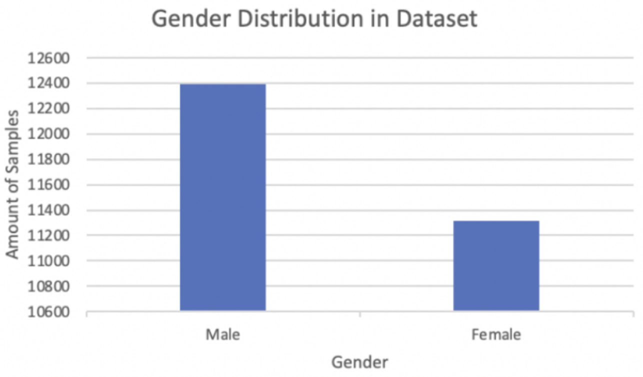

Fig. 3. Gender split in the dataset

B. CNN Architecture

Our CNN architecture is comprised of 5 layers. There are 2 convolution layers, 2 fully connected layers, and 1 output layer which can be seen in Figure 4. The CNN is an end to end sequential deep learning architecture. In addition, we also used LeakyReLu for our activation functions and batch normalization in both convolution and fully connected layers.

Each classification/regression problem has their own unique output layer where age regression had 1 output node, while gender classification had 2 nodes, and ethnicity classification has 5 nodes.

In the beginning we had three CNNs with two convolutional layers, one fully connected layer, and one output layer. Adding an additional fully connected layer increased the accuracy of our models. We also changed over from ReLU to leaky ReLU. This allows the architecture to keep learning even if the gradients become small, because the gradients can never be 0.

Layer Type	**Output Size**	**Filter size/stride**
Input Image	48 x 48 x 1	———
CONV1	44 x 44 x 32	5 x 5
Maxpool1	22 x 22 x 32	2 x 2 / 2 x 2
CONV2	18 x 18 x 64	5 x 5
Maxpool2	9 x 9 x 64	2 x 2 / 2 x 2
FC1	1000	———
FC2	100	———
Output	1 / 2 / 5 (age / gender / ethnicity)	———

TABLE I
A TABLE OF THE LAYERS INCLUDING MAXPOOLS.

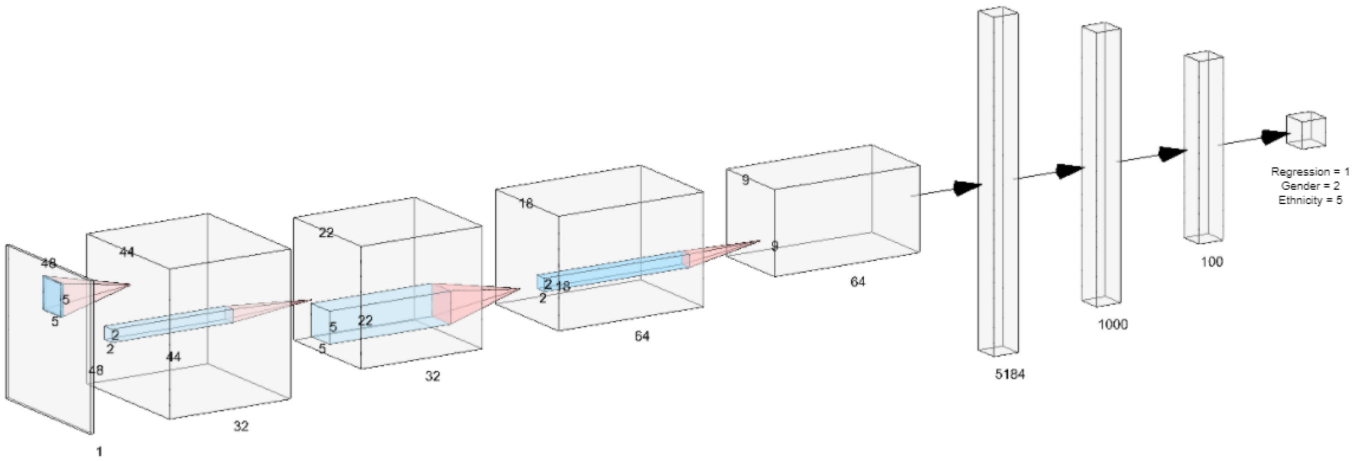

Fig. 4. The architecture of the age, gender, and ethnicity CNNs

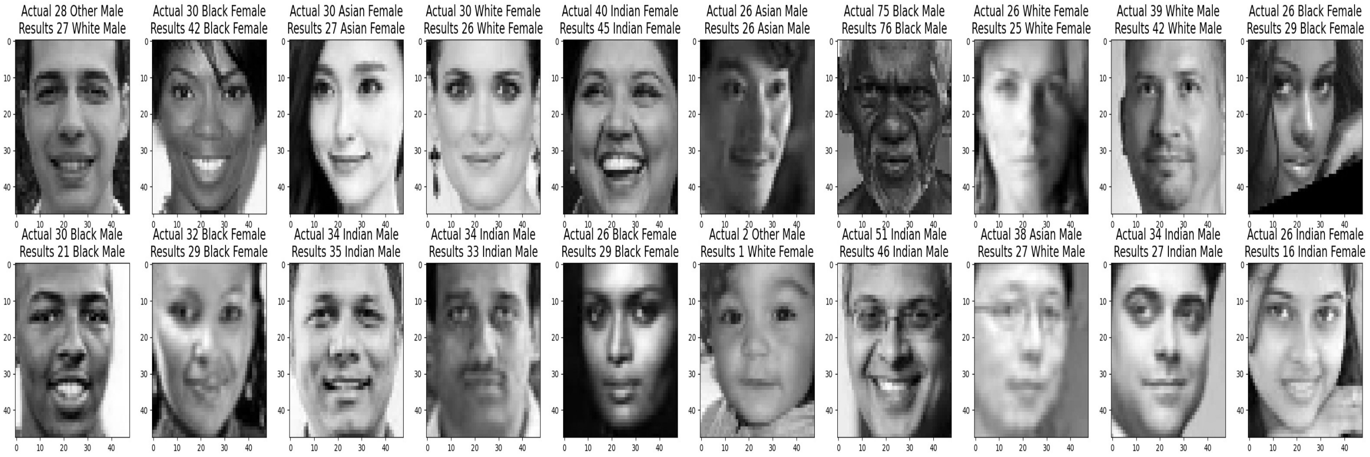

Fig. 5. Predictions from the Age, Gender, and Ethnicity CNNs for a subset of images.

III Results and Analysis

1) Results

The samples were taken from the test set so that none of these images have been seen by the model. The majority of the samples seem reasonable and are classified correctly, but there are some misclassifications. An example of a misclassification is where the actual ethnicity is Other, the ethnicity model predicts that the person is White. This is seen in the picture in Row 1 Column 1 of Figure 5, and the image in Row 2 Column 6. There was also a misclassification in Row 2 Column 8 where the person is Asian but the model predicted the ethnicity as White. The model leaning towards predicting White may have something to do with the face that the dataset's majority ethnicity was White.

The age predictions for this subset of images usually are close, being within 1-3 years of the actual age of the subject. In some cases though, the prediction is not very close to the actual age. An example of this is in Row 2 Column 10, where the woman is actually 26 but the model predicted she was 16. This is also seen in the Row 2 Column 8 image where the man is 38 but the model predicts he is 27. Also, since the images are small (48 x 48) the images did not contain a lot of detail. For some of the age predictions, this could have been a deciding factor when fine details such as wrinkles may not have been present in the sample image.

For this subset, two gender predictions were wrong. This was for the infant in Row 2 Column 6, and for the man in Row 2 Column 8. The age of the infant may have made it harder for the model to predict his gender correctly. It is hard for humans to tell the gender of infants so it makes sense that it is harder for the model to predict the gender accurately as well.

Overall, the model does a surprisingly good job in predicting the age, gender, and ethnicity of the person. The model could have been impacted by the resolution of the images, and the dataset it trained on.

A. Accuracies and Losses

1) Age CNN

The accuracy for age seen in Figure 6 was low because it is difficult for a regression task to predict the exact number. However, the loss was around 70 which is good since the range of ages are 1 to 116. Note that the accuracy for the age model is also including a ±1 age allotment.

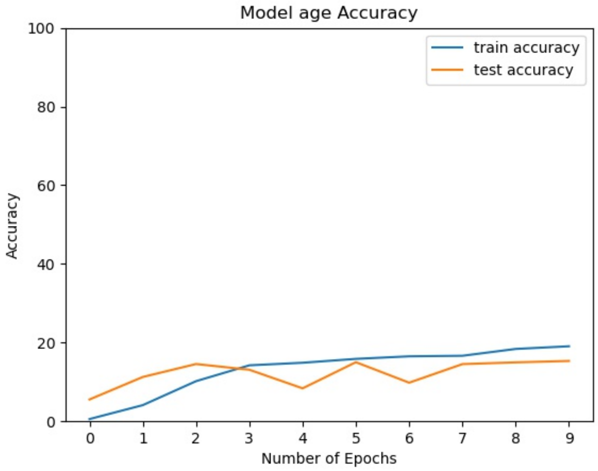

Fig. 6. Accuracy per epoch for the age CNN

The loss curve is an ”L” shape in Figure 7 which means that the learning rate is appropriate for the model, and the age model is not overfitting or underfitting. Since the test loss goes down with the train loss, it looks like that’s the best this type of model can do.

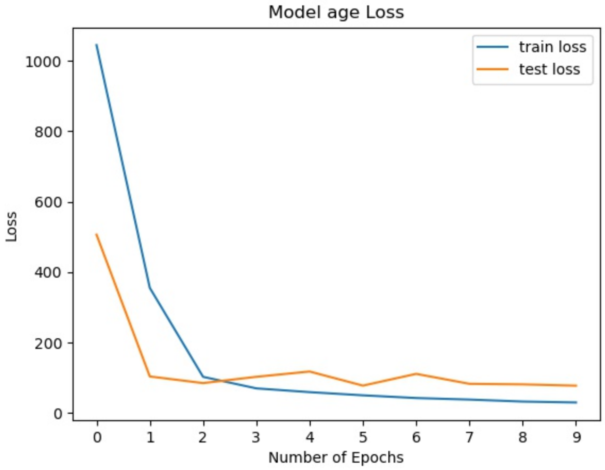

Fig. 7. Loss per epoch for the age CNN

2) Gender CNN

The accuracy for the gender is hovering around 91 percent in Figure 8 and even through training keeps getting better, the test accuracy plateaus at a lower accuracy. Yet the model is still performing well.

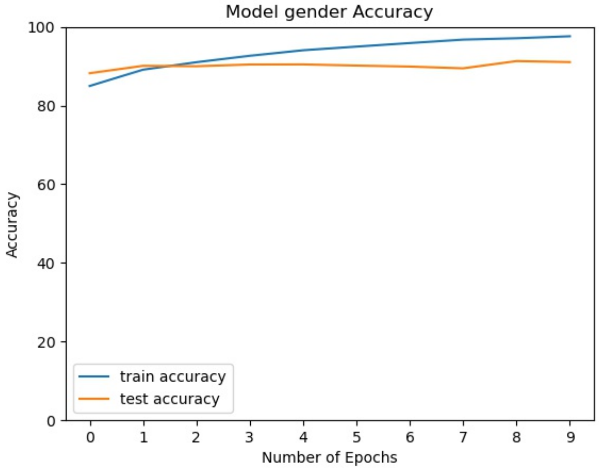

Fig. 8. Accuracy per epoch for the gender CNN

Regarding loss, the test loss starts increasing after epoch 4 which can be seen in Figure 9 due to the model overfitting to the training dataset so we saved the model with the lowest test loss and used that for inference. The train loss goes down as the epochs increase, which is desired. But the train loss is similarly not an ”L” shape which means that the learning rate may be too low for this model.

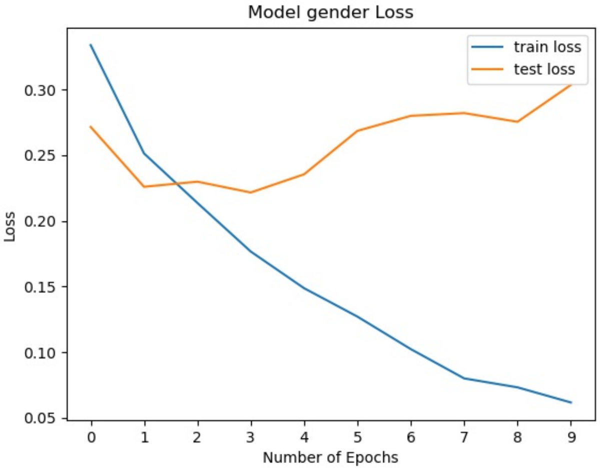

Fig. 9. Loss per epoch for the gender CNN

3) Ethnicity CNN

The accuracy for the ethnicity is hovering around 80 percent and even through training keeps getting better the test accuracy can only get so good.

4) Accuracy and Loss

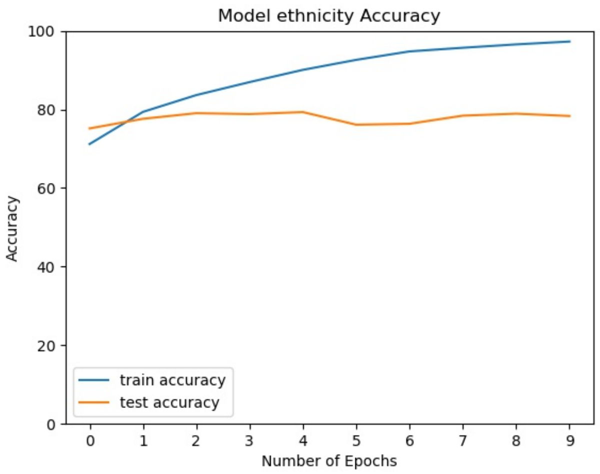

Fig. 10. Accuracy per epoch for the ethnicity CNN

As we can see in figure 11 the loss for the test loss starts increasing significantly after epoch 4 due to the model overfitting to the training dataset so we saved the model with the lowest test loss and used that for inference.

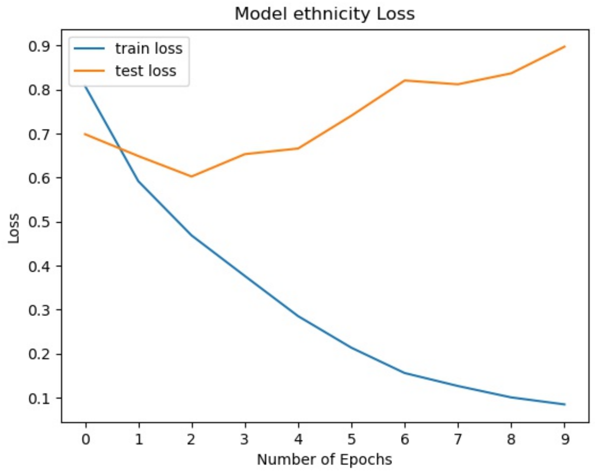

Fig. 11. Loss per epoch for the ethnicity CNN

IV Discussion

The CNNs that we created performed well given the amount of layers and the dataset it was trained on. The gender predictions had the highest accuracy, this may because there were only two categories it had to predict (male or female). The ethnicity predictions were second best, when the CNN had a choice between White, Black, Asian, Indian, and Other. The age prediction was understandably had the worst performance of the three, but usually the age model predicted close to the actual age of the subject.

If the project had more time, instead of getting the dataset from Kaggle, we could have preprocessed the dataset ourselves directly from the UTK dataset in order to have higher resolution of either 96x96 or a 224x224 resolution samples. The higher resolution would allow for us to use pre-train models since the resolution would be the same as the images those pre-train models were trained on. In addition, it would help with determining age since winkles and fine lines would be visible at that resolution. The channel of rgb values would be kept to preserve colors instead of just a gray scale image. This could affect the predictions of the CNN, because color plays an important role in predicting ethnicity and age.

After getting a high accuracy we would look into a multi headed model instead of a separate model for each specific classification/regression task. In addition we might incorporate a little bit of down-sampling from the majority classes to see if that helps the model have a higher accuracy in prediction.

It would also be interesting to spend more time to further quantify the bias in the age, gender, and ethnicity CNNs. The ethnicity CNN for the subset of results that we examined tended to predict ethnicities such as White incorrectly. Further experiments could be run to determine how the dataset sample amounts for each category effect the accuracy for test samples of the same categories. This could give greater insight into how big of a role the dataset plays in the quality of a CNN's output.

V Conclusion

Overall, we learned a lot during this project. We learned the most from experimenting with the layers to try to get better results. Of course, there could be improvements to the models, as discussed in the discussion section. Yet, we were successful in creating a CNN that produces results that are similar to the related work in this area of research. The CNNs had an 80% accuracy in ethnicity and 91% accuracy in gender.

VI Contributions

Both George Lu and Sarah Wilson worked on the entire project together. The work was 50/50, it included preprocessing, creating the model, training, and writing the paper.

References

[1] Nipun Arora. Age, gender and ethnicity (face data) csv. https://www.kaggle.com/nipunarora8/age-gender-and-ethnicity-face-data-csv/version/1, 2020.

[2] Carnegie Endowment for International Peace. Ai global surveillance index. https://carnegieendowment.org/files/AI$_{G}lobal_{S}urveillance_{I}ndex1.pdf$, 2019.

[3] IBM. Convolutional neural networks. https://www.ibm.com/cloud/learn/convolutional-neural-networks, 2020.

[4] Gil Levi and Tal Hassncer. Age and gender classification using convolutional neural networks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 34–42, 2015.

[5] Yang Song and Zhifei Zhang. Utkface large scale face dataset. https://susanqq.github.io/UTKFace/, 2017.