RSNA Bone Age

Individual Project Report

Akhil Kumar Baipaneni

Introduction

Bone age is an interpretation of skeletal maturity, typically based on radiographs of the left hand, fingers, and wrist. Generally, the bone age of a healthy child may be a year or two advanced or delayed. A child's bone age is determined by comparing the bones in the x-ray with the bones of a standard atlas, usually "Greulich and Pyle". Many factors influence the progression of skeletal development, including nutrition, genetics, hormones, and disease states.

In this project, we focus on bone age (in months) prediction using x-ray images of children's left hand. We are using Convolutional Neural Networks for this purpose. The image must be pre-processed before passing it to the model. TensorFlow is my choice for the framework. We used pretrained model Xception which worked best for us. Below are the details of the dataset

Source: https://www.kaggle.com/kmader/rsna-bone-age

Total x-ray images: 12611

Individual Work

I wrote pre-processing, training and prediction code for my model. I tried training multiple different custom architectures, with differing number of layers of different types and with many different shapes. I didn't get better performance on the testing data. After these bad results, I decided to try out pre-trained models.

As part of feature engineering, I created a new column age class by converting continuous bone age values into 10 bins and then plotted a histogram on that column to see the age distribution in each category. From the plot, I observed that there are less values in some of the bone age categories. So, for splitting the dataset into train, test and validation sets, I have used age class for stratified sampling. I have created a new column bone_age_z with normalized bone age values. The images are resized to 256 * 256 pixels.

Then I performed image augmentation with the below configurations to have different variations in the training images.

- Rotation range = 20 (I have taken less rotation range value as there will not be much rotation in the x-ray images)
- Width shift range = 0.2
- Height shift range = 0.2
- Shear range = 0.02 (Took less value as I don't want to do much scaling on x-ray images)
- Zoom range = 0.3
- Fill mode = nearest
- Horizontal flip = True (To have flipped x-ray images)
- Brightness range = [0.5, 1] (For increasing the brightness of the images)
- Data format = channels_last (As I was having the channels in the last index)
- Preprocessing function = Xception preprocess_input module

For modelling, I have used Xception pretrained model with include top layer as False and trainable as True. That means I am unfreezing the layers for the weights to get updated during training. This will form my base model. Then I proceeded with fine tuning the fully connected layer (top layer) by taking the output of the base model as input. I have added three layers with 512, 10 and 1 neuron with ReLU activation in the first two layers and Linear in the last layer. The models are evaluated using the mean absolute error which is calculated in months.

Results

I have used AWS for training the models. The models are trained using a batch size of 32 images. I had issues in allocating memory (ran out of memory) while running the code. Learning plateaus at 40 epochs and this takes about 4 hours. Adam is used as an optimizer with a learning rate of 1e-3. I have also implemented early stopping with patience value as 30. The pre-trained model Xception with fine-tuned top layer work decently which gave a validation absolute error of 9.36 months. That means the predicted bone age value with be having a deviation of 9.36 months from the actual value.

```
Epoch 37/200
253/252 [==============================] - 312s 1s/step - loss: 0.0873 - mae_in_months: 9.5760 - val_loss: 0.0911 - val_mae_in_months: 9.6506
Epoch 38/200
253/252 [==============================] - 311s 1s/step - loss: 0.0832 - mae_in_months: 9.3826 - val_loss: 0.0937 - val_mae_in_months: 9.8625
Epoch 39/200
253/252 [==============================] - 311s 1s/step - loss: 0.0834 - mae_in_months: 9.3657 - val_loss: 0.1090 - val_mae_in_months: 10.5230
Epoch 40/200
253/252 [==============================] - 311s 1s/step - loss: 0.0824 - mae_in_months: 9.3180 - val_loss: 0.0996 - val_mae_in_months: 10.2073
Epoch 41/200
253/252 [==============================] - 311s 1s/step - loss: 0.0782 - mae_in_months: 9.0573 - val_loss: 0.0913 - val_mae_in_months: 9.8012
Epoch 42/200
253/252 [==============================] - 312s 1s/step - loss: 0.0749 - mae_in_months: 8.7894 - val_loss: 0.0852 - val_mae_in_months: 9.3645
```

Summary

Pre-trained model clearly outperforms the custom CNN models.

We can further improve this model by correct pre-processing of the images like the parameters in the data augmentation if we have proper domain knowledge. Also, in future, I am planning to ensemble of models to get better results.

Code proportion:

18-10/ (18+130) = 5.40

References:

https://github.com/yuxiaohuang

https://pediatrics.aappublications.org/content/140/6/e20171486