Bone age prediction from X-ray images of hand

Chirag Soni (1401CS13), Vignesh Edithal (1401CS14), Prashant Singh (1401CS35)

April 8, 2018

Abstract

Pediatricians use skeletal age of children to ensure their normal growth by comparing it with actual age. This project attempts to predict skeletal age from X-ray images of hand and the gender of the child. We used the dataset from RSNA pediatric bone age challenge hosted on Kaggle. Gender information is crucial for age prediction, we verify this by comparing various models. The best result was obtained using the InceptionV3 model provided with Keras. We compare various deep image models and perform hyper-parameter optimization on them.

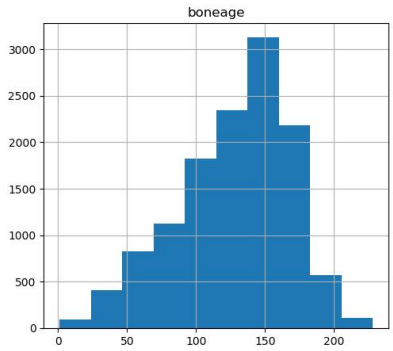
1 Introduction

The Radiological society of North America (RSNA) had organized a challenge in 2017 for pediatric bone age identification using machine learning. Hand x-rays are performed in pediatric patients to detect growth abnormalities by comparing their skeletal age with the actual age to ensure it is within normal limits. While automated approaches to bone age analysis have previously been developed and are commercially available today, none are widely available and radiologists are stuck with the task of flipping through the Greulich and Pyle atlas to find the most similar example every time they are presented with a bone age study. With such a cumbersome and dated method, bone age analysis is one of the "low hanging fruits" of medical imaging in this renaissance of artificial intelligence.

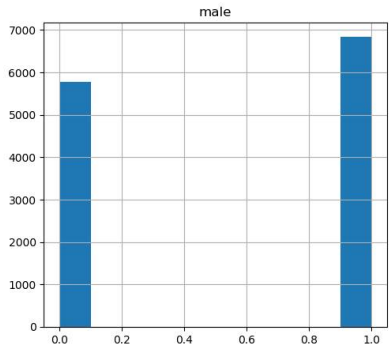
We had a dataset of 12,611 X-ray images which was hosted on Kaggle. Although the dataset was not large enough for training deep neural networks, yet we were able to predict with satisfactory error margins. This project attempts to use deep learning image models to predict the skeletal age using X-ray images of hand and compares it with that estimated by radiologists. Theoretically gender of a person is crucial for skeletal age identification due to different growth rates in both the genders (typically 2 years difference is observed), we explore the validity of this by comparing different models. We use some pre-trained image models available with Keras, augment it with our own components and perform hyper-parameter optimization to finally get a low mean absolute error. The code is uploaded on Github .

2 Dataset

We used the dataset provided by the RSNA for the competition comprising of 12,611 x-ray images of hand . These images came with labels for their skeletal age in months and the gender of the patient in a separate csv file. As the dataset has age as well gender information so we decided to train a neural network that takes into account gender instead of training two separate neural networks for each gender. Skeletal maturity can vary significantly by gender, up to as much as 2 years as females mature much faster than males. Omitting gender for the network architecture would impact the model's performance. We split the available data into training, validation and testing datasets by randomly choosing 72%, 8% and 20% of the instances respectively.



(a) Age distribution in the dataset.



(b) Gender distribution in the dataset.

Figure 2: Sample X-ray image from the dataset.

The distribution of age and gender is visualized using the following histograms.

3 Pre-processing of the data

Our image sizes were about 2000 x 1500 pixels (3 MP) in 8 bit greyscale format, but we used resized images of resolution 224 x 224 pixels in 8 Bit RGB format as it is the standard inputs for standard deep image models such as InceptionV3, MobileNet, VGG16 etc...
Keras provides the ImageDataGenerator class that defines the configuration for image data preparation and augmentation, we use it to generate generalized images. For the training set, we used a rotation range of 5 degrees, horizontal/vertical translation up to 15%, zoom up to 25% and a horizontal flip. Through a bit of experimentation we found the following parameters to be optimal.

```
horizontal_flip = True, height_shift_range = 0.15, width_shift_range = 0.15,
    rotation_range = 5, shear_range = 0.01, zoom_range=0.25
```

The data image generator returns batches of manipulated images when requested. Image Augmentation is the process of taking images that are already in a training dataset and manipulating them to create many altered versions of the same image. This both provides more images to train on, but can also help expose our classifier to a wider variety of lighting and coloring situations so as to make our classifier more robust. By doing this, we force the network to learn features which are intrinsic to the patient rather than the imaging technique. Images of a child's hand can easily be acquired with varying zoom, rotation, position as well as either the left or right hand and these

factors do not impact our abilities as radiologists to analyze the image.

4 Architecture

We have experimented with Inception V3, VGG16, Mobilenet models for our task. Keras provides these models out of the box with pre-trained weights. We extracted the layer after the final concatenation layer from the Inception V3 network, applied Global Average 2D Pooling on it , fed it to a dropout layer with 0.2 dropout. We created gender network which takes as input binary gender information (0 for female or 1 for male) and fed it through a 32-neuron densely connected layer. Output of both these layers were concatenated . The concatenated layer was then fed through two additional 1024-neuron densely connected layers with 'relu' activation and dropout layers after each of the dense layers. The final layer consists of a single neuron with linear activation which predicts the age.

The purpose for using 32 neurons in the gender network is to balance the relative contribution of each input (pixels and gender) into the final decision. At our concatenation layer, the pixels contribute 2048 inputs while the gender contributes 32. We picked this ratio because we did not want to bias the network too significantly based on gender input, but we wanted to give it the ability to impact the overall prediction.

The additional dense layers give the network more parameters to learn in order to adjust during training to allow it to reason out the relationship between the pixel and gender information.

A single numeric output rather than separate classes for each month was more intuitive and came with the added benefit of avoiding similar classes activating together.

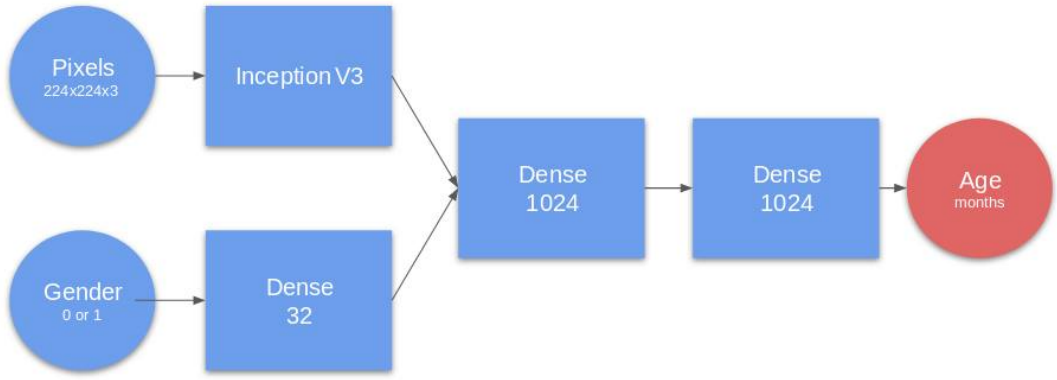


Figure 3: Model Architecture

4.1 Hyperparameters

Optimization was done by varying the following hyper-parameters, and comparing the results.

- Pooling : Average pooling / Max pooling
- Dropout : 0.2 / 0.5
- Activation function : ReLU / Tanh
- Gender : Use or not
- Image Size : 224x224, 500x500
- Pretrained models : Inception V3, VGG, MobileNet

4.2 Training

Training was done with following settings:

- Training Batch size of 10 instances
- Trained for a maximum of 50 epochs
- Adam optimizer
- Mean square error as loss function
- Mean absolute error as additional metric
- Reduce Learning Rate on Plateau
- Early stopping with patience of 10 epochs in case the validation loss does not improve

5 Results

We generated a total of 8 models by varying the hyper-parameters such as dropout value, activation function, incorporating gender information, image size, using pre-trained weights on imagenet dataset and adding an extra hidden dense layer of 1024 neurons before the final layer. The specification of the models are as follows:

- **Model 1**
  - InceptionV3 pretrained network used
  - Global Average 2D pooling used after output of InceptionV3 network
  - Gender network is **not** used
  - Hidden layer activation: **Tanh**
  - Dropout = 0.5
  - Image Size = (224,224)
  - Number of hidden layers = 1
- **Model 2**
  - MobileNet pretrained network used
  - Global Average 2D pooling used after output of CNN
  - Gender network is **not** used
  - Hidden layer activation: **Tanh**
  - Dropout = 0.5
  - Image Size = (224,224)
  - Number of hidden layers = 1
- **Model 3**
  - InceptionV3 pretrained network used
  - Global Average 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Tanh**
  - Dropout = 0.5
  - Image Size = (224,224)
  - Number of hidden layers = 1
- **Model 4**

  - InceptionV3 pretrained network used
  - Global Average 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Relu**
  - Dropout = 0.2
  - Image Size = (224,224)
  - Number of hidden layers = 1

- **Model 5**
  - InceptionV3 pretrained network used
  - Global Max 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Relu**
  - Dropout = 0.2
  - Image Size = (500,500)
  - Number of hidden layers = 1

- **Model 6**
  - VGG16 pretrained network used
  - Global Average 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Relu**
  - Dropout = 0.2
  - Image Size = (224,224)
  - Number of hidden layers = 1

- **Model 7**
  - InceptionV3 network with **randomly initialized weights** used
  - Global Average 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Relu**
  - Dropout = 0.2
  - Image Size = (224,224)
  - Number of hidden layers = 2

- **Model 8**
  - InceptionV3 pretrained network used
  - Global Average 2D pooling used after output of CNN
  - Gender network is used
  - Hidden layer activation: **Relu**
  - Dropout = 0.2
  - Image Size = (224,224)
  - Number of hidden layers = 2

Performance Metrics of various models					
Model No.	Train Loss	Train MAE	Valid Loss	Valid MAE	Test MAE
Model 1	0.4827	21.9844	0.3689	19.1613	19.449
Model 2	0.2349	15.4782	0.1865	13.7690	26.3639
Model 3	0.3227	17.8639	0.1702	12.7672	13.083
Model 4	0.0889	9.4319	0.0736	8.4977	8.842
Model 5	0.2878	16.2868	0.1143	10.8331	11.2478
Model 6	2.4902	56.4327	1.1138	36.9110	35.309
Model 7	0.1609	12.7924	0.1073	10.6136	11.0849
Model 8	**0.0809**	**9.0737**	**0.0709**	**8.1923**	**8.615**

The highlighted row is the best performing model with a mean absolute error of 8.6 months on the testing dataset. Approximate training time per epoch was about 420 seconds on Nvidia GTX 1080 Ti GPU.

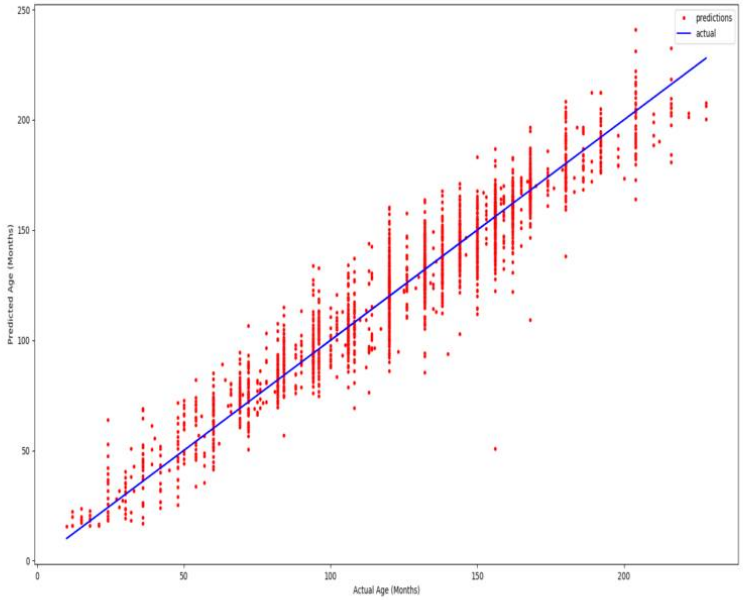


Figure 4: Predicted Age Vs Actual Age

Figure 5: Sample Results

6 Conclusion

Larson et al. performed extensive statistical analysis on the interobserver difference for the images in the provided test set which was independently reviewed by three additional pediatric radiologists. They found that the MAD between a reviewer and the average of the other reviewers ranged between 0.53 to 0.69 years (6.36 to 8.28 months) with a mean of 0.61 years (7.32 months).

The winning model in the competition gives a MAD of 0.67 years (8.1 months). This proves that AI today is capable to match or sometimes easily beat human efficiency. The question now arises is that should we stop training radiologists? In our opinion, we should not. Radiologists will be essential for the future of AI in medicine as they are best positioned to identify, direct, and implement AI to solve the most impactful clinical problems facing medical imaging today.

Instead of heralding our demise, intelligent tools will broaden our reach and impact as a specialty by improving our efficiency and helping us to maintain the same high level of accuracy and quality. Machine learning, deep learning, artificial intelligence will be the foundation of these next-generation tools and, ultimately, will allow us to provide faster, better, and more reliable care to our patients.

7 References

- Kaggle RSNA Bone Age Dataset.
- RSNA Challenge.
- Performance of a Deep-Learning Neural Network Model in Assessing Skeletal Maturity on Pediatric Hand Radiographs .
- Lee, H., Tajmir, S., Lee, J. et al. J Digit Imaging (2017) 30: 427. Fully Automated Deep Learning System for Bone Age Assessment .