Appendix E1: 2017 RSNA Machine Learning Challenge Organizing Committee and Sponsors (alphabetical order by last name)

1. Kathy Andriole, Massachusetts General Hospital and Brigham and Women's Hospital Center for Clinical Data Science

2. Bradley J Erickson, Mayo Clinic

3. Adam E Flanders, Thomas Jefferson University

4. Safwan Halabi, Stanford University

5. Jayashree Kalpathy-Cramer, Massachusetts General Hospital

6. Marc Kohli, University of California-San Francisco

7. Luciano M Prevedello, The Ohio State University Wexner Medical Center

Appendix E2: Challenge Timeline

Challenge site opening: August 1, 2017

Training data phase: August 1 to 30, 2017

Leaderboard phase: September 1 to October 7, 2017

Availability of the test dataset: October 7, 2017

Submission of results: October 7 to 15, 2017

Review and confirmation of results: October 15, 2017

Notification of awardees: October 30, 2017

Public announcement of results: Monday, November 27, 2017 at the RSNA Annual Meeting Machine Learning Pavilion.

Appendix E3: Challenge Terms, Conditions, and Rules

Participants in the Challenge were judged by how well the bone age evaluations produced by their algorithms predicted the expert assessment. Participants had the opportunity to directly compare their algorithms in a structured way using this curated dataset. The RSNA Machine Learning Challenge Organizing Committee selected a small group of the most successful entries for recognition at the RSNA Annual Meeting. Recognition of Challenge participants is part of a broad range of educational events and exhibits focusing on machine learning at the RSNA Annual Meeting.

The goal of the Challenge was to develop an algorithm using ML techniques which most accurately predicted skeletal age based on pediatric hand radiographs. Participants submitted CSV files with the case ID and the predicted age in months. A prediction was required in all cases in each phase to be counted. A ZIP file containing this CSV file as well as a text file describing the methodology used to determine bone age was uploaded as the participants' or teams' submission.

The primary evaluation of each ML algorithm measure was Mean Absolute Distance (MAD) in months, calculated as the mean of the absolute values of the difference between the model estimates and those of the reference standard bone age. The concordance correlation coefficient (CCC) was selected as a secondary measure to solve potential ties.

Other terms and conditions:

1. Entrants agree to hold harmless the organizers and their institutions, as well as RSNA and its staff and contractors for any costs, harm or damage incurred in the course of participating.

2. Entrants retain all rights to algorithms and associated intellectual property they develop in the course of participating in the challenge.

3. Entrants may make identified or pseudonymous submissions. A valid e-mail must be associated with each submission and will be used to communicate with entrants.

4. Results are to be submitted as a zip folder containing a single .csv file of submission data and an MS-Word or text document describing your research methods.

5. Entrants may make multiple submissions up to the deadline date. The MedICI system will acknowledge successful submissions and provide error messages to help entrants make any fixes needed in submitted data sets.

6. A data board will be made public on the day the challenge ends (Oct. 15).

7. Entrants are required to submit a brief statement (2–3 paragraphs) describing the methods they used, including the size of the data set required for prediction. These statements may be used by the organizing committee in developing general publications about the challenge.

8. Entrants may use only publicly available models trained on publicly available data sets. Entrants may not use other external data sets.

9. Entrants may reannotate images in the training set, but may not have human observers rate and evaluate the test data set.

10. Entrants may use transfer learning. However, these models must be trained only on publicly available data (eg, GoogleNet).

11. The availability of training and test data sets will adhere to the following schedule:

a. Training Data Available: Aug. 5

b. Training Phase: Aug. 5-Sept. 1

c. Leaderboard Data Available: Sept. 1

d. Leaderboard Phase: Sept. 1-Oct. 7

e. Test Data Available: Oct. 7

f. Submission of Test Results: Oct. 7-Oct. 15

g. Review and Confirmation of Results and Notification of Awardees: Oct. 15

h. Acknowledgment of Awardees at RSNA Annual Meeting: Monday, Nov. 27

12. Awardees may be offered the opportunity to participate in research publications led by the Organizing Committee. Entrants may publish other research papers publications based on their participation in the Bone Age Challenge only with approval of the Organizing Committee.

13. We ask that you personally abide by codes of honor and ethical behavior in your participation. Only one registration is allowed per participant and compliance will be monitored. During the Test phase, starting on October 7, 2017, registration to the site will be closed and only existing participants/teams will be able to submit results (maximum of 3 submissions). Team creation will also be disabled during the Test phase. We view this competition as a unique opportunity to further contribute to the field of machine learning in medical imaging and we appreciate your most legitimate efforts to make this activity a great success.

14. Organizers of the challenge and groups that provided the data sets used are not allowed to participate as entrants in the challenge.

Appendix E4: Challenge Platform

The Challenge was hosted on the MedICI platform (built on CodaLab) provided by Jayashree Kalpathy-Cramer and Massachusetts General Hospital, supported through NIH grants (U24CA180927) and a contract from Leidos (Leidos Health, Westfield, Indiana, USA).

The Challenge website: http://rsnachallenges.cloudapp.net/competitions/4

Appendix E5: Top Ten Results with their corresponding competition usernames, nationality (top five) and MAD Score (months)

1. 16 Bit Inc. (Canada)/4.265
2. Ian Pan (USA)/4.350
3. Felipe Kitamura (Brazil)/4.382
4. Hans Thodberg (Denmark)/4.505
5. MD.ai (USA)/4.527
6. amiper/4.555
7. rsnahandchallenge/4.561
8. grin/4.802
9. lbicfigraz/4.881
10. jcrayan/4.907

The data and results of all the participants' submissions can be accessed at this interactive website: http://rsnachallenges.cloudapp.net:5006/rsna_interactive

Appendix E6: Detailed description of 16 Bit Inc. approach and model (First Place)

Data and Preprocessing

The provided data were split into an 85:15 training:validation split resulting in 10,720 training images and 1,892 validation images. Input pixel dimensions of 500 × 500 was selected as an optimal size for the problem, provided dataset, and the available GPU memory.

Model Architecture

Both gender and pixel information were incorporated into a single network using the inception v3 architecture for the pixel information as shown in Figure 3. The layer after the final concatenation layer from the Inception V3 network was extracted, flattened, and concatenated with the gender network which took as input binary gender information (0 for female or 1 for male). Prior to concatenation, the gender input was fed through a 32-neuron densely connected layer. Following concatenation, two additional 1000-neuron densely connected layers with 'relu' activation were used before the final single-output layer.

The motivation for this design stems from the relative contribution of each input (pixels and gender) into the final decision. At the concatenation layer, the pixels contribute 100,384 inputs while the gender contributes 32. This ratio was selected so as to not bias the network significantly based on gender input, yet still allow for the gender to impact the overall prediction. The additional dense layers provide the network with more trainable parameters to learn of the relationship between the pixel and gender information. A single numeric output rather than separate classes for each month was more intuitive and comes with the added benefit of avoiding similar classes activating together.

Training

Keras 2.08 with TensorFlow 1.3 using Python 3.4 was used. Over 40 experiments were performed on 2 machines, one housing an NVIDIA P40 and two Titan X GPUs and another housing a single Titan X GPU. No pretraining or data or model parallelism was used.

Real-time data augmentation on the entire set was performed using rotation range of 20 degrees, horizontal/vertical translation up to 20%, zoom up to 20% and a horizontal flip. This was done to force the network to learn features which are intrinsic to the patient rather than the imaging technique.

The final model was trained with a minibatch size of 16 for 500 epochs (approximately 50 hours) with the ADAM optimizer attempting to minimize mean absolute error of the output. The learning rate was reduced when the validation loss plateaued. The best models achieved MAD of 5.^{99}Months on the validation set.

Inference

Many papers reference a 10-crop-validation scheme wherein predictions on random crops of the test samples are averaged to reduce the effect of outlier predictions (5,8). A similar scheme was devised and implemented with a Keras generator using horizontal/vertical translation up to 25% with horizontal flip. The generator yielded 10 samples for each of the top 5 models resulting in

50 predictions for each test image. Results were averaged and rounded to the nearest integer to arrive at the final prediction.

The 16 Bit Inc. bone age model is available for public use and testing at this website: https://www.16bit.ai/bone-age

Appendix E7: Detailed description of Ian Pan's approach and model (Second Place)

Data and Preprocessing

We manually cropped images such that the hand comprised the majority of the image and then resized such that the length of each image was 560 pixels, while maintaining the original aspect ratio. A length 560 pixels was selected after validating the performance of several other lengths: [448, 672, 896, 1008]. We applied contrast-limited adaptive histogram equalization (CLAHE) to enhance bone borders (Fig 4). From each image, we extracted 49 patches of size 224 × 224 pixel, with each patch centered on a point in an evenly-spaced 7 × 7 grid over the image. The patch size was chosen to match the native input size of the ImageNet pretrained ResNet50 network. Each patch was used as input to the neural network.

Model Architecture

We selected the ResNet50 CNN architecture with pretrained ImageNet weights for our prediction model and trained separate models for males and females. The network was modified to accept grayscale input.

Training and Validation

We split the data into 10 folds (90% training, 10% validation). Model training and evaluation were performed using the Keras 2.0 API and the Python 2.7 programming language on 2 NVIDIA GTX 1080 Ti GPUs. We trained models on randomly sampled image patches in subepochs of $n = 16{,}000$. We first trained the fully connected output layer for 10 subepochs using the Adam optimizer with learning rate 0.01. We then fine-tuned up to the second identity block (Keras layer 37) for 100 subepochs using Adam, using an initial learning rate of 0.0002 for 20 subepochs. At 20 subepochs, we halved the learning rate; the learning rate was halved every 10 subepochs thereafter. We used real-time data augmentation with probability 0.5 for regularization. One of 4 transformations was randomly selected and applied to an image selected for data augmentation: (1) Gaussian blur with random sigma between 0.2 and 1, (2) left-right flip, (3) random clockwise rotation between -30 and 30 degrees, and (4) random magnification between 1.11× and 1.33 ×. We validated models every 5 subepochs and selected the model with the lowest validation error. The above procedure was repeated 3 times for 3 out of 10 folds (9 models per sex).

Inference

The Xth percentile of the predictions for an image's 49 patches was used as the final prediction, where X was tuned on a validation set. × was on average around 50, or the median of the distribution. To combine image predictions from the 3 models within each fold, we calculated a weighted average of the 3 predictions for each patient, where weights were selected from [(1,1,1), (2,1,1), (3,1,1)] and the model with the lowest validation error was given the highest weight. The averaged predictions from each of the 3 folds were then averaged once more using a simple mean to generate the final prediction for each patient.

Appendix E8: Detailed description of Felipe Kitamura's team approach and model (Third Place)

Data and Preprocessing

Due to the high variability of the aspect ratio and the resolution of the images in the dataset, our team decided to resize them. After some experimentation we reached an optimal format of 550 × 550 pixels. We kept the original aspect ratio of the images and padded them symmetrically with zeros.

Model Architecture

When developing our neural network, our team had two main insights:

1-Concatenate the gender input with the extracted features:

We concatenated the gender input to the features extracted by our convolutional layers and added an extra fully connected layer to allow some nonlinearity to the final estimator (Fig 5).

2-Novel architecture module:

We proposed a new module that uses a transpose convolution followed by a combination of a convolution and a pooling operation, in parallel with a residual connection.

Transpose convolutions have been widely used in semantic segmentation neural networks like FCN (9) and U-Net (10). These networks have an encoder-decoder architecture and the transpose convolution operation is used in the decoder part of the network. The operation is used to upsample the feature maps generated by the encoder, but instead of doing a simple interpolation they are trainable and can learn a more efficient nonlinear transformation.

Combined with skip connections, the upsampled feature maps reach higher spatial resolution. Since the bone age assessment task relies on the detection of small details in the radiographs, we assumed a CNN used in this task could benefit with less coarse feature maps, hence the introduction of the new module.

The module was named the ice module in reference to the fire module introduced in the SqueezeNet (11). The fire module first compresses and then expands the feature maps' information in each layer of the SqueezeNet architecture. It also acts on the number of feature maps (depth-wise). Our module does the opposite. It first expands and then compresses the feature maps' spatial information to achieve a more refined feature representation (Fig 6).

We introduced two of these modules into a seven-layer vanilla CNN along with Batch Normalization layers to help reduce training time and improve the regularization of the network. This network was used to compose our final submission and it has around 1% in number of parameters when compared with Inception v4 or ResNet-152.

We are still investigating the properties of the module and how it can benefit other computer vision tasks. We expect to publish a deeper analysis in the near future.

Training

Training was done with random online data augmentation in Keras, on top of TensorFlow, using a Titan X GPU.

Inference

The final output is the average of the best 4 models from Cross Validation with five folds. We discarded the model that overfitted the most.

Appendix E9: Detailed description of Hans Thodberg's approach and model (Fourth Place)

Data and Preprocessing

The Active Shape Model (ASM) (12) and the Active Appearance Model (AAM) (13) are used to locate the contours of 15 bones as shown in Figure 6. One ASM is developed for each bone. The ASM and AAM are classic machine learning techniques: they implement a deformable template that automatically warps to fit each bone. The training data consists of 50 or more examples of bones which have been annotated by 64 or more landmarks on its boundary. The Minimum Description Length (MDL) method (an unsupervised machine learning method) (14) was used to partly automate the annotation. The ASM consists of two components: One is a statistical shape model which is trained by unsupervised learning from the annotated example; it is basically a principal component analysis. This allows the ASM to know the plausible bone shapes. The second component is a method for locating each landmark in the images-this can either be learned from the training data, or it can be a rule-based concept, eg, the location of the strongest gradient. A particular challenge with ASM and AAM is the need to have good initial guesses of the location of the bones. This is solved by performance first an exhaustive search for the characteristic metacarpal 2–5 bone group all over the image, in all rotations, in a range of magnifications and for both left and right hands configurations, and picking a set of most likely locations as a basis for coarse-to-fine searches for the metacarpals. This is in turn used for searches for the remaining 11 bones. This preprocessing works equally well for all orientations of the hand and automatically copes with the approximately 1% right hands in these data.

Training and Validation

Bone age is estimated on each of 13 bones using "hand-crafted features," as opposed to features learned by deep learning (Fig 7). Three kinds of features are used: The shape of the bone, the intensity pattern across the growth zone and the pattern of Gabor texture energies across the growth zone. These features are processed in two step. The first step is unsupervised learning by Principal Component Analysis, and the second step is linear regression against the observed bone age, followed by a polynomial transformation. Separate models are made for bones with and without separate epiphyses. The predictions from the 15 bones are averaged to produce the final prediction of bone age. The method estimates 300 parameters for each bone from the 12611 training cases. Reducing the training set size to 2000 cases increases the MAD on a test by only 6%, so this method does not require a lot of data. The method was first presented in (15).

Appendix E10: Detailed description of Leon Chen's team approach and model (Fifth Place)

Data and Preprocessing

Segmentation mask: about 400 random images were manually annotated with masks for the hand/wrist/distal forearm. These annotations were used to train a dilated convolutional U-net for predicting segmentation masks at a resolution of 512 × 512. This network was used to create segmentation masks for the remaining unannotated images. It also serves as the first module in the overall pipeline.

Input data are preprocessed by first predicting the hand/wrist/distal forearm segmentation mask using the trained segmentation neural network, then cropped, scaled, and padded to a resolution of 299 × 299. Input images consist of 3 channels: raw cropped/scaled/padded x-ray, corresponding segmentation mask, and masked x-ray. Data augmentation applied during training includes random rotations (± 40 degrees), random x-y shifts (± 75 pixels), random scaling (80%–120%). 90/10 training/validation split on the data were used, with a mini-batch size of 32.

Model Architecture

Libraries:

TensorFlow, Keras, Numpy, Scipy, Pandas, Pillow, OpenCV

Hardware:

Nvidia Titan X

Bone age prediction:

Multiple convolutional neural networks were trained with mean absolute error loss using Adam with a decreasing learning rate schedule. These networks contain a final regression layer (sigmoid activation, scaled to output 0–228), with gender information incorporated through an embedding layer, which is multiplied with the penultimate layer prior to the final regression layer. The entire network is then trained end-to-end. Training was carried out until validation error stopped decreasing, typically about 200 epochs.

Inference

Final predictions were created by an ensemble (using a simple weighted average) consisting of networks with the following model architectures: ResNet-50 with global average pooling, Inception-V3 with global average pooling, Xception with global average pooling, Xception with global max pooling, Inception-ResNet-V2 with global average pooling, Inception-ResNet-V2 with global max pooling.

Appendix E11: Comparison and discussion of the top five methodologies

In the description of the five methods above, three methods used an explicit preprocessing step:

-Ian Pan manually cropped the images to standard size and location

-Thodberg segmented the hand into 15 bones

-MD.ai used a convolutional net to segment the hand using an overall mask

The other two methods, however, did not reduce the location and size variation. Instead, they used data augmentation to impose such variations on all training cases. It is interesting to see that these two strategies lead to a similar performance.

In general, the variability among the five methods is large. All methods formulate the problem as a regression problem (rather than a classification problem), but apart from that, the diversity is remarkable. One method uses conventional (ie, nondeep) machine learning, while Kitamura used a 7 layer CNN with Ice Modules and the others used deeper CNNs, with a preference for ResNet and Inception. Only Pan used transfer learning. Still, the spread in performance is quite small. It is interesting to ask whether the performance on the test data allows us to state that some methods are better than others.

To present a quantitative analysis of this, we performed, as an example, a comparison between two of the methods: Pan and Thodberg. The difference in MAD between these is 0.14 months. To decide whether this is a statistically significant difference, we used the standard "bootstrapping technique": We sampled 200 examples from the 200 test cases with replacement, and did this 100000 times, and for each sampling, we computed the MADs of the two methods. In 71% of the samples, the Pan methods performed better. We can adopt the convention that a method must be better in 95% of the samples to constitute a significantly (or robustly) better method (reference DREAM challenge). This corresponds to $P < .05$ in a paired, one-sided test. In order for the Pan method to be significantly better than the Thodberg method, its MAD must be shifted so that it is 0.46 month better than the other method. If we assume that this margin also holds when comparing any other pair of models, we can state that all methods with a MAD within 0.46 months of the winner method are not significantly inferior to the winner. This margin includes to top seven methods, so the choice of the top five as "winners" was a fair approximation.

All five methods used some kind of ensembling, ie, averaging or otherwise combining predictions from several different sources to form the final prediction:

-16 Bit used the average of 5 models, each used on 10 translations and horizontal flips of the images

-Pan used the median of 49 predictions made on different subimages and a 9-model average

-Kitamura used the average of four models

-Thodberg used the average predictions from 13 bones

-MD.ai used the average of six different CNN architectures

These ensemble techniques are "within-laboratory." They reflect some arbitrariness in the predictions that obviously should be averaged out.

Ensembling can be taken further (16). In this challenge, we have several rather different methods with approximately the same performance, and this makes it sensible to ensemble different solutions. To illustrate the power of this, we formed the average of the Pan and Thodberg methods, and found this ensemble to have a MAD = 4.001 months, ie, surpassing the winner. The theory behind this is the following: For each method, consider the residuals of the predictions, ie, the differences between the predicted and the true values. To measure how similar two methods are, we use the correlation R between their residuals. If the two methods perform approximately equally well, the average of the two is expected to have an RMS errors, and therefore also a MAD error, which is sqrt ((R+1)/2) times the individual MAD. Between the Pan and Thodberg models we have R = 0.61, ie, these methods are indeed rather different, and we expect the MAD of the ensemble to be 0.89 times the individual MAD, in good agreement with what we observed.

Finally, we formed the average of all five methods-the MAD of this ensemble was approximately 3.8 months. So, progress is this area could come not only from optimizing each method, but as much from collecting ensembles of diverse methods developed by independent teams, therefore reinforcing the importance of challenges such as this.

To place these performance numbers (the MADs) in context, we first remark that these five methods obtained MAD in the range 6.1–6.5 months in the leaderboard phase, where the ground truth was a single radiologist's reading, while they achieved 4.3–4.5 months in the test phase, where the ground truth was the average of six raters. Thus, the rater variability had a major contribution in the leaderboard performances.

References

9. Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

10. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation—medical image computing and computer-assisted intervention. Arvix: 1505.04597 [preprint]. https://arxiv.org/abs/1505.04597. Posted May 18, 2015. Accessed November 3, 2017.

11. Iandola FN, Han S, Moskewicz MW, Ashraf K, Dally WJ, Keutzer K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and ,0.5MB model size. Arxiv:1602.07360 [preprint]. https://arxiv.org/abs/1602.07360. Posted February 24, 2016. Accessed November 3, 2017.

12. Cootes TF, Taylor CJ, Cooper DH, Graham J. Active shape models: their training and application. Comput Vis Image Underst 1995;61(1):38–59.

13. Cootes TF, Edwards GJ, Taylor CJ. Active appearance models. In: Burkhardt H, Neumann B, eds. Computer Vision — ECCV'98. ECCV 1998. Lecture Notes in Computer Science, vol 1407. Berlin, Germany: Springer, 1998; 484–498.

14. Thodberg HH, Olafsdottir H. Adding curvature to minimum description length shape models. Proceedings of the British Machine Vision Conference, Vol 2, 2003; 251–260.

15. Thodberg HH, Kreiborg S, Juul A, Pedersen KD. The BoneXpert method for automated determination of skeletal maturity. IEEE Trans Med Imaging 2009;28(1):52–66.

16. van Ginneken B, Schaefer-Prokop CM, Prokop M. Computer-aided diagnosis: how to move from the laboratory to the clinic. Radiology 2011;261(3):719–732.