

Online Supplemental Information

METHODS

Convolutional Neural Network Algorithm. For image classification tasks, the convolutional neural network models utilized cross-entropy loss. During model training, the model learns the correct classification of audiogram images through iterative attempts by minimizing loss. If the model produces an incorrect label, the loss value is high and is back-propagated through the network to minimize future labelling losses. We found that re-training ResNet using simply the default parameters and settings obtained from fast.ai (i.e., using the default cross-entropy loss function with a recommended range of learning rates obtained using the fast.ai ‘Learning Rate Finder’ function) performed excellently on our training and held-out data and that we did not require any specific hyper-parameter tuning or cross-validation to improve model performance. Only the training set was used for model training and for the learning rate finder. The model validation performance outcome metric was classification accuracy which compared the model prediction against the audiogram human expert label of the validation data only.

Fast.ai Module Hyperparameter Settings. Accuracy in image classification tasks using CNNs can be influenced by model hyperparameters including image batch size, number of training epochs and the learning rate. Under the recommendation of the fast.ai development team, our model trained under the 1-cycle strategy that varies learning rates in the early epochs from low to high to low.¹ Prior to model training, an optimal learning rate was determined, and a range of learning rates were retained for the 1-cycle strategy. This strategy reduces overfitting and prevents model learning from stagnating in a local error/loss minimum.² The

default value of 30 epochs, and a batch size of 24 was used.

Image Augmentation Settings. To generate permutations of training data to boost overall training set size, we permitted the fast.ai data loader (see <https://docs.fast.ai/vision.transform.html>) to mutate the images through random horizontal image flipping (no vertical), random lightning and contrast change (setting = 0.2), random image rotation to a maximum of 35 degrees, a random zoom of 1.05 (or 105%), and random symmetric warps (setting = 0.2).

Hardware & Software Environment. The model training and analyses were completed on a custom-built desktop computer with a AMD Ryzen Threadripper 1950X processor, 32GB of read-only memory (RAM), and a GeForce RTX 2070 8GB graphics card. Data processing and neural network generation was completed using the ‘fast.ai’ (www.fast.ai, San Francisco, California) PyTorch deep learning environment running in an Ubuntu Linux operating system.

Supplemental Methods References

1. Smith LN. *A disciplined approach to neural network hyper-parameters: Part 1 -- learning rate, batch size, momentum, and weight decay*; 2018. <http://arxiv.org/pdf/1803.09820v2>.
2. Smith LN, Topin N. *Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates*; 2017. <http://arxiv.org/pdf/1708.07120v3>.