Voice Accent Learning Using Recurrent Neural Networks on Spectrograms

Henry Wang
henryfw@stanford.edu
https://youtu.be/OXLvXS1NTPI

Introduction

- The goal is to be able to learn the difference between the American accent and the British accent, so that the difference can be used to partially alter the voice of someone with an American accent with a British accent

- The model uses is a many-to-many two-layer LSTM network with two fully connect networks at each time slice

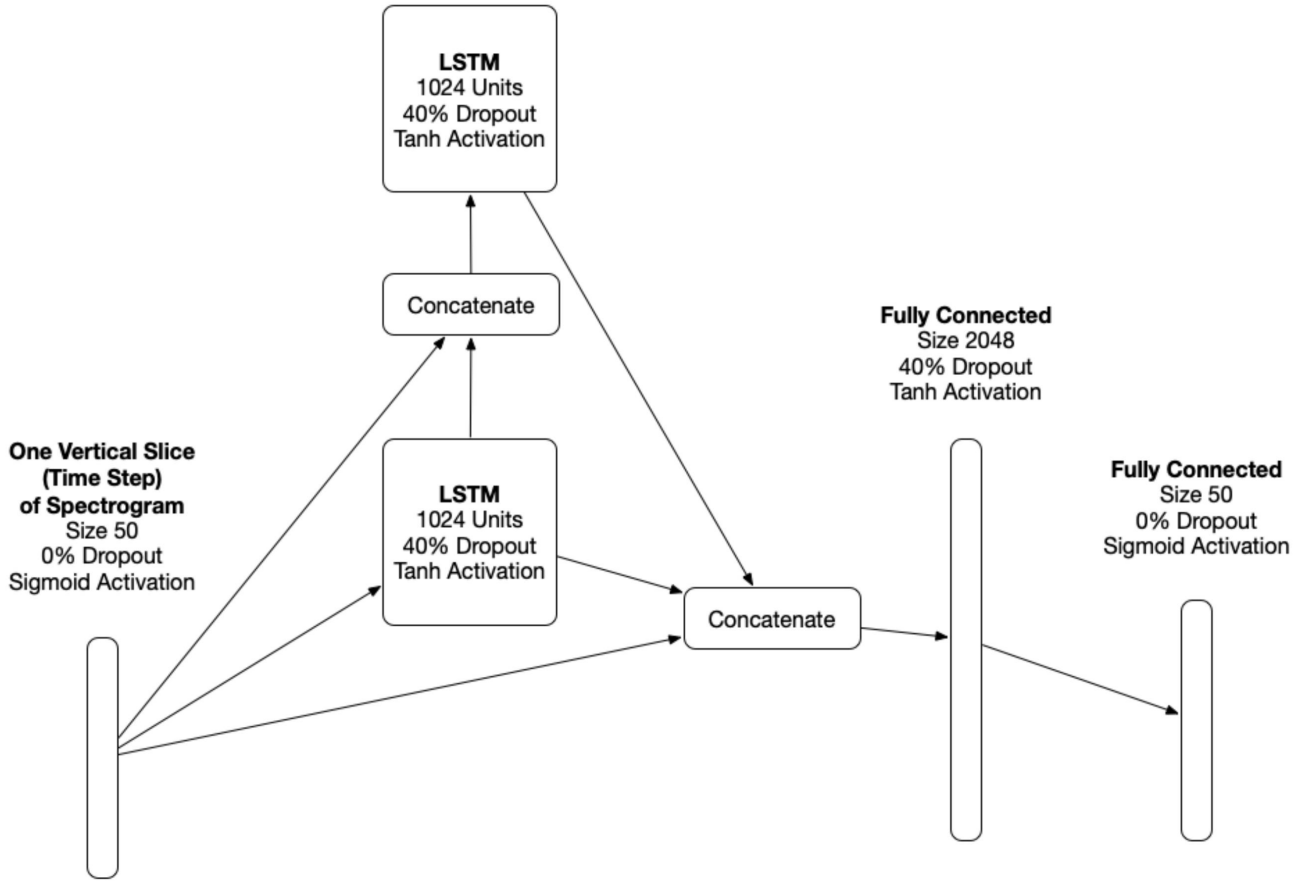
- Input data are spectrogram images of computer synthesized speech in American accent and British accent

- Results show that the model works when trained with input data consisting of one word at a time

Data

- Single words spoken using computer generated voices in American accent and British accent
- WAV sound files are converted to spectrogram images cropped to 150 (x-axis for time) by 50 (y-axis for pitch)
- Values are normalized to 0 and 1.0 and used as features in this time-series dataset, which is appropriate for training seq-to-seq RNN models
- Words are mined from 90k IMDB movies reviews with more frequent words repeated up to 5 times
- Total dataset is 350k pairs of spectrogram images in randomized order

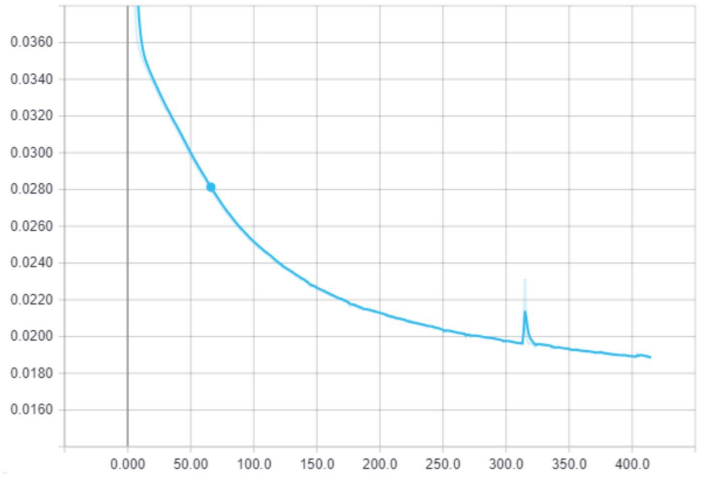
RNN Model


LSTM
1024 Units
40% Dropout
Tanh Activation
Concatenate
Fully Connected
Size 2048
40% Dropout
Tanh Activation
One Vertical Slice
(Time Step)
of Spectrogram
Size 50
0% Dropout
Sigmoid Activation
LSTM
1024 Units
40% Dropout
Tanh Activation
Concatenate
Fully Connected
Size 50
0% Dropout
Sigmoid Activation

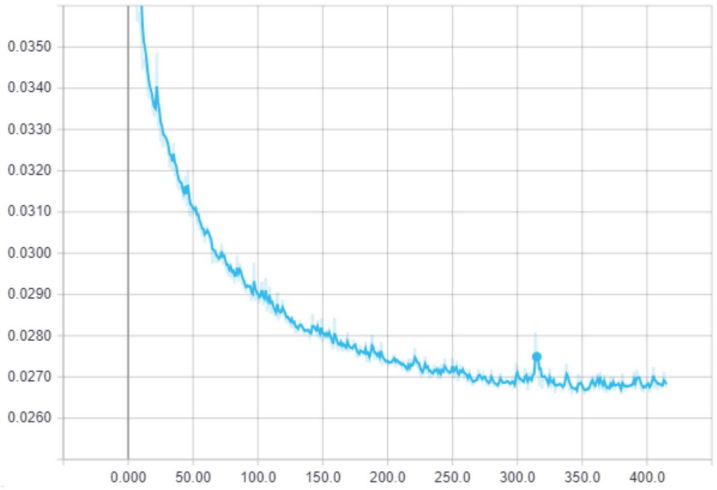
Training

- Keras and Tensorflow used
- Loss function is mean absolute error
- Optimizer is RMS Prop
- 40k pairs of spectrograms are used in training and 5k pairs are used in validation
- Final model trained on NVIDIA 1070 GTX for 400 epochs over 40 hours
- Training loss is 0.019 and validation loss is 0.027

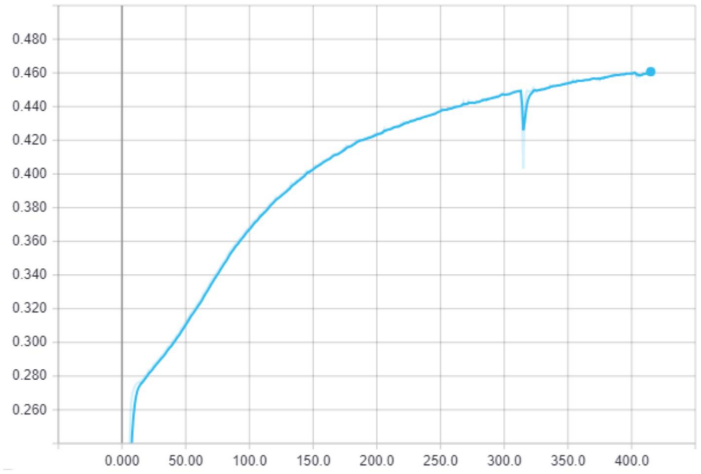
Training Loss



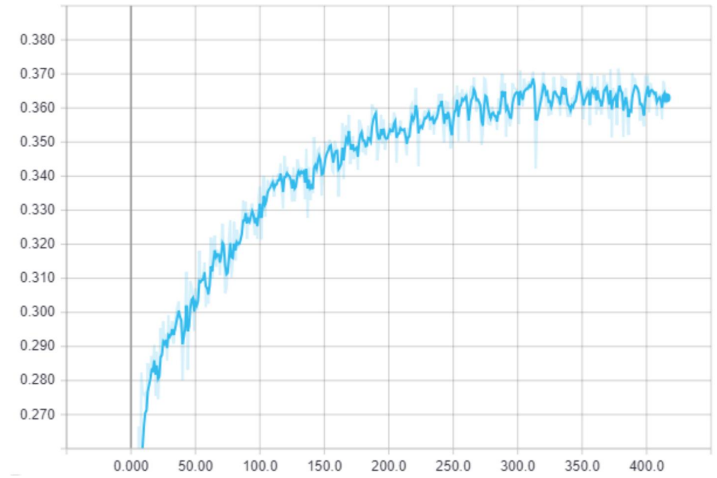
Validation Loss

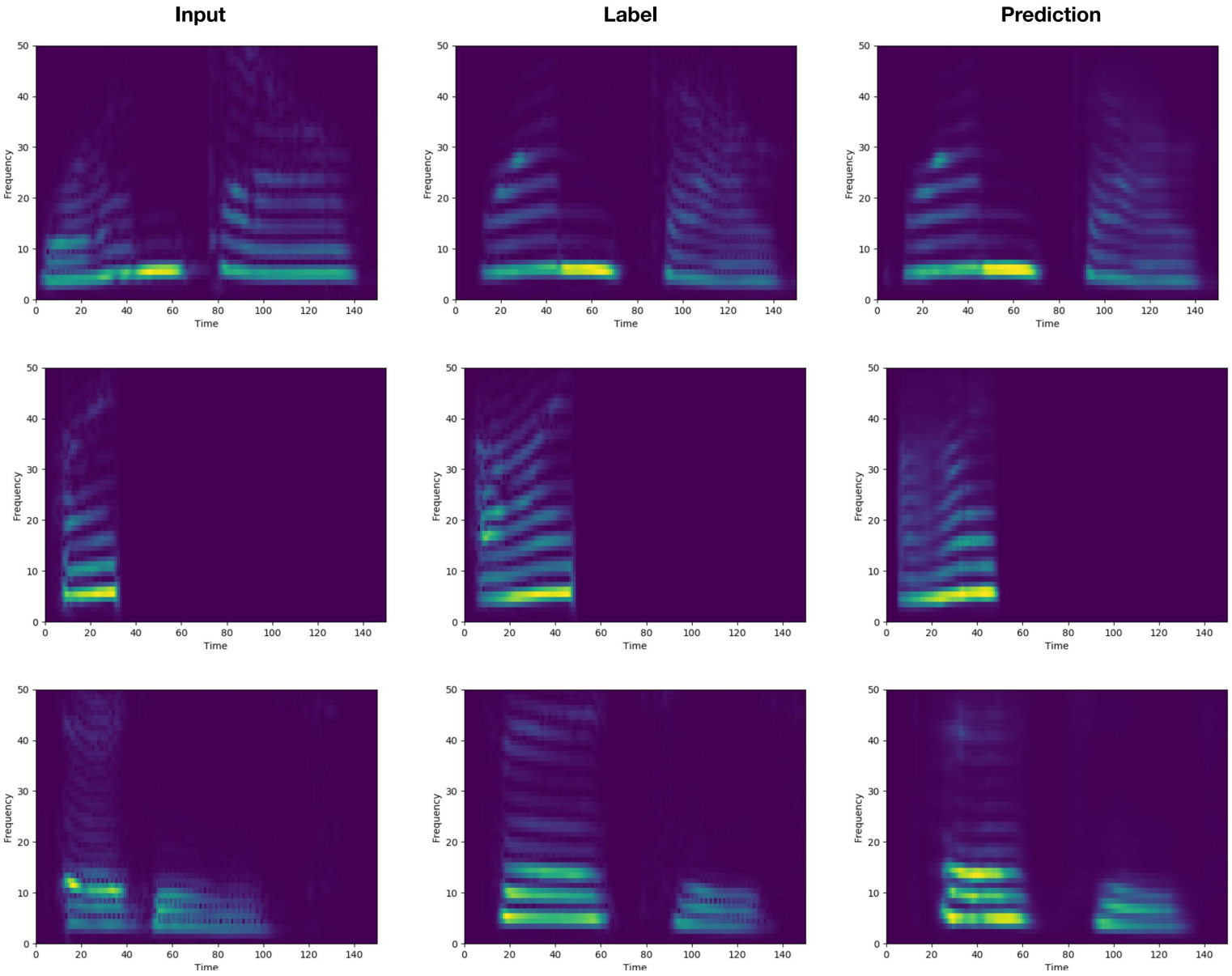


Training Accuracy



Validation Accuracy




Input
Label
Prediction
Frequency
Time

Discussion

- The two-layer recurrent neural network setup is usually used as the encoder and decoder layers in language translation. It did moderately well for the translation of spectrograms in two accents. The results are due to the powerful abilities of recurrent neural networks to remember sequences, even when each step is a large vector.

- Almost all the layers received the input in addition to the immediate previous layer. These skip connections help the back-propagation steps during training, especially when using tanh activations.

- The tanh activation did much better than relu, which produced unstable results.

- Dropouts are very helpful in preventing overfitting with respect to the validation loss.

- Without the intermediate size 2048 FC layer, the initial results were not worth pursuing.

- Using only one LSTM of 1024 units layer did not work as well initially, so was not pursued.

- Training takes a long time, measured in days, if not weeks for more realistic datasets.

Future

- This project only trains one single word at a time within a fixed temporal window. This means pre-processing and post-processing will be required to chop sentences into words, which can be future work.
- Due to time and resource constraints, the full spectrograms of 300x257 were cropped to 150x50. Most of the intensities in pitch occurred in this area, but future work could be done on the entire image.
- While this project only used 40k samples to train, future work can use all of the 350k generated samples.
- Greater vocabulary and additional voice pairs can be used in the future.
- Float16 is used in this project, but future work should probably use float32 or greater to prevent underflow with larger training samples.
- A model with more LSTM units can also be experimented with in the future.
- Subjectively test using a real human speakers.

https://youtu.be/OXLvXS1NTPI