URL to Colab:

https://colab.research.google.com/drive/1frxIi7TEYTZHeWqHjR05ZBsex7aT95OP

Project Explanation:

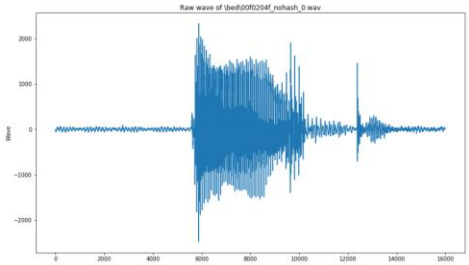
Speech recognition is widely used in our lives, especially mobile apps or Internet of Things. This notebook is to utilize Kaggle speech recognition challenge dataset to create a Keras model on top of Tensorflow and make predictions on the voice files.

Data Ingestion and Processing:

Similar to image recognition, the most important part of speech recognition is to convert audio files into 2X2 arrays.

Sample rate and raw wave of audio files

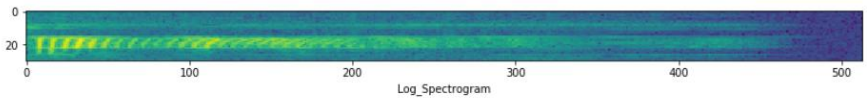
Sample rate of an audio file represents the number of samples of audio carried per second and is measured in Hz. The following image shows the relationship between the audio raw wave and sample rate of “bed” audio file:



Spectrograms

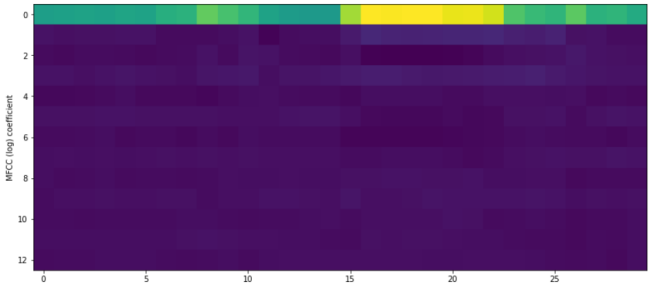
The spectrogram is a spectro-temporal representation of the sound. The horizontal direction of the spectrogram represents time, the vertical direction represents frequency. (1)

Spectrograms can be used as a way of visualizing the change of a nonstationary signal’s frequency content over time. The following image shows the log of the spectrogram of audio “bed”:



Mel Frequency Cepstral Coefficient (MFCC)

Mel Frequency Cepstral Coefficents (MFCCs) are a feature widely used in automatic speech and speaker recognition. The Mel scale relates perceived frequency, or pitch, of a pure tone to its actual measured frequency. Humans are much better at discerning small changes in pitch at low frequencies than they are at high frequencies. Incorporating this scale makes our features match more closely what humans hear. (3)
The following image shows the MFCC of audio “bed”:



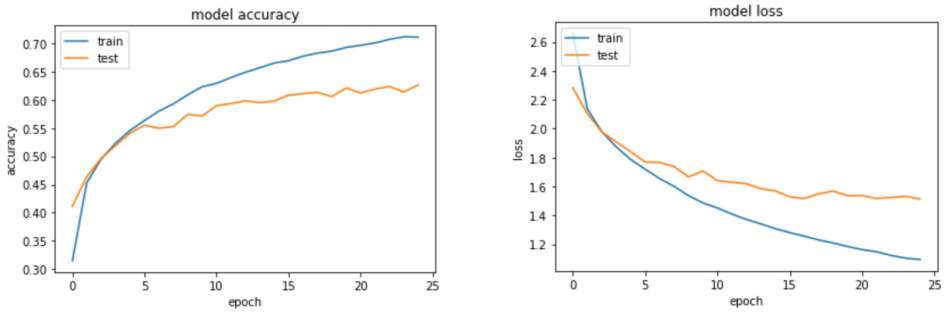
Data Modeling:

I built a sequential neural network model, which is the simplest way to build a model in Keras. Then I added four dense layers, which are fully connected layers in the model.

After building the model, I used Adaptive Moment Estimation as optimizer, categorical crossentropy as loss, and accuracy as metrics to compile the model.

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 30, 32)	448
dense_2 (Dense)	(None, 30, 64)	2112
dense_3 (Dense)	(None, 30, 128)	8320
flatten_1 (Flatten)	(None, 3840)	0
dense_4 (Dense)	(None, 30)	115230
activation_1 (Activation)	(None, 30)	0

I chose 25 as the number of epochs, which is the number of times the model will cycle through the data. After running around 20 epochs, the validation accuracy of the model is improved to 61% - 62%.



As we can see from the above two pictures, the test and train accuracies are not close enough to each other, which means this model can be improved by overcoming overfitting problems.

Project Conclusion:

1. Audio file is usually converted to array to be the input of Keras model.
2. Spectrogram and MFCC are the two features of audio files to be converted to arrays.
3. We can modify the layers of Keras model to increase the model accuracy.
4. Be aware of overfitting problems by comparing training and testing accuracies.
5. Sequential model is easier to modify, compared with Keras API model.

Reference

1. https://www.kaggle.com/ibtesama/getting-started-with-a-movie-recommendation-system/data
2. https://hackernoon.com/introduction-to-recommender-system-part-1-collaborative-filtering-singular-value-decomposition-44c9659c5e75
3. http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/