Speech Assignment 1

Name: Amgad Mohamed Ahmed Abdo

ID:20201446888

Objective:

Given a dataset containing cat and dog sounds it is required to make a model that classifies input to either a dog or a cat.

Steps:

Step1:Data Collection and preprocessing

We download dataset from Kaggle website and upload it on google drive, then we download Zip file from drive onto google colab notebook and unzip it.

After reading both train and test datasets we check the if the training data is balanced, we discover that the data skews more for cats than dogs, so we remove some cat labeled samples for training data to be balanced and shuffle both training dataset and test dataset, due to the small size of the dataset we opted out of using a validation split as it would have caused training dataset to be much smaller therefore less information could be gathered for the model.

We create a function that will extract wavform from its destination and the label given to it, then we decode and squeeze wavform data so it would be suitable for the next step, we apply this on our training data and prepare it for the next step of preprocessing.

After getting the waveform we pass our data to a function that transforms the audio data from wavform to Spectrogram that can be used in the model, we also change each label into numerical form of {0,1} instead of {cat, dog} so we can pass it to model.

We apply all previous techniques on test dataset then splitting training data into batches of 64.

Step2:Building and training model

Using tensor flow we create a model that consists of a normalization layer to normalize the dataset, 8 convolutional layers with 3 filters each and each one uses ReLU activation function, these layers are used to extract the features from input spectrogram, after we use max pooling and dropout layer to extract features into a smaller window size and avoid overfitting respectfully, then we use a dense layer of SoftMax activation function, dropout layer and another dense layers that contains the probabilities of the labels.

```
Input shape: (2726, 129, 1)
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resizing (Resizing)         (None, 32, 32, 1)         0

 normalization (Normalizati  (None, 32, 32, 1)         3
 on)

 conv2d (Conv2D)             (None, 30, 30, 32)        320

 conv2d_1 (Conv2D)           (None, 28, 28, 32)        9248

 conv2d_2 (Conv2D)           (None, 26, 26, 32)        9248

 conv2d_3 (Conv2D)           (None, 24, 24, 32)        9248

 conv2d_4 (Conv2D)           (None, 22, 22, 32)        9248

 conv2d_5 (Conv2D)           (None, 20, 20, 32)        9248

 conv2d_6 (Conv2D)           (None, 18, 18, 32)        9248

 conv2d_7 (Conv2D)           (None, 16, 16, 32)        9248

 max_pooling2d (MaxPooling2  (None, 8, 8, 32)          0
 D)

 dropout (Dropout)           (None, 8, 8, 32)          0

 flatten (Flatten)           (None, 2048)              0

 dense (Dense)               (None, 128)               262272

 dropout_1 (Dropout)         (None, 128)               0

 dense_1 (Dense)             (None, 2)                 258

=================================================================
Total params: 327589 (1.25 MB)
Trainable params: 327586 (1.25 MB)
Non-trainable params: 3 (16.00 Byte)
```

In this model we use Adam optimizer with a learning rate of 0.0001 and using Sparse Categorical Cross entropy to calculate the loss, the model is trained on 500 epochs.

Step3:Evaluation of model

We pass test dataset through the model and achieve accuracy of 85% then we create confusion matrix for test data that shows as follows:

```
3/3 [==============================] - 0s 10ms/step
Test set accuracy: 85%
```