# Udacity-ML-Capstone Proposal
# Dog Barking Detection

By Mike Lam
March 2nd, 2018

## Domain Background

It is an Acoustic Event Detection (AED) classification task, but it can be turned into an image classification problem. Convolutional Neural Networks (CNN) have proven to be very effective in image classification and show promise for audio set. [1][2]

The transformation process requires knowledge of audio signal processing. Frequency of the audio is very important for identifying and classifying the source. Since in real life many signals are not stationary, the frequency domain will be changing through time. Turning the signal into time-frequency domain is a very popular technique. This is called time-frequency analysis and is very popular. [3]

## Problem Statement

Puppies bark a lot for various reasons, for instance they might be scared, might be hungry, might hear someone walking past the house, etc. Owners would want to know if their dogs or puppies are barking, again for many reasons. The motivation of this project is for owners who cannot spend daytime with their puppies because of work and have to leave them in crate. Through detecting barking instances, a barking session can be defined as a period of consecutive barking. Then owners can know how many barking sessions happened during the day. This is a classification problem with 10-second audio inputs. The output is whether it is a dog barking or not (one versus all classification).

## Datasets and inputs

Google AudioSet [4] is a sound vocabulary. It consists of an expanding ontology of 632 audio event classes and a collection of over two-million human-labeled 10-second sound clips from YouTube videos. The ontology ranges from human and animal sounds to musical instruments and genres, and common everyday environmental sounds. Each video is labeled one or more classes. For this project, each label will be turned into bark or not bark.

The audios are stored on YouTube. A spreadsheet with the urls to the target videos is provided. Videos are uploaded by different people so there is no standard sampling rate. The whole dataset is imbalanced because only a small portion is dog barking videos. For barking videos, there are 0.2 hour of videos as training set and another 0.2 for evaluation set. The evaluation set will be split into validation and testing set evenly. There is another 7 hours of videos of which the quality is worse and imbalanced. In this project, several sizes of training data will be tried. Smaller and more balanced dataset will be tried, and the size is increased and more imbalanced dataset will be used. Two fixed size relatively balanced (50-50 or 40-60) dataset will be used for validation and testing.

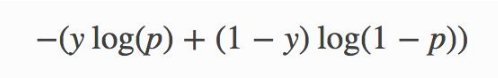
The raw input will be a 10-second audio stored in wav file. The audio file is then pre-processed. The final input will be a series of frames of time-frequency signal in the form of images. The labels will be changed to bark/not bark. Then the problem will change from audio event detection to an image binary classification task. [1]

## Solution Statement (proposed solution)

I intend to apply CNN to this problem. Barking of dogs and puppies should have a specific pattern which should be reflected in the time-frequency plot. Similar patterns should be observed in those plots when barking is occuring. As CNN is good at uncovering the abstract features that can describe the image, it is a good choice of algorithm to tackle this problem. Training CNN from scratch might not achieve good results because this dataset is not huge compared to YouTube-8M dataset [5]. Transfer learning will be applied in an attempt to overcome this issue.

## Benchmark Model

A fully connected neural network will be used as the benchmark model. It will take flattened inputs from the images of time-frequency plot and output whether the given series of frames contain a bark. This model can contrast with the ability of CNN model that can retain and process spatial information. Cross entropy [6] will be used as the loss function, given by the following formula.



$$-(y \log(p) + (1 - y) \log(1 - p))$$

## Evaluation Metric

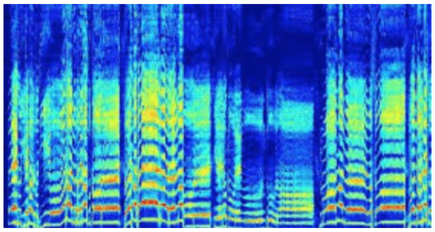
Since we are only interested in knowing whether a bark has occured, this image classification task is a binary classification problem, bark or not bark. This one-versus-all situation would result in an imbalanced dataset where majority is not bark. The evaluation metric used is F1 score [7], which is the harmonic mean of recall and precision. Precision measures how many predictions made are correct, while recall measures how many positive test samples are picked up. F1 is a balanced measure of the two. A good classification model should be able to correctly classify and not missing out correct ones, hence a high F1 score.

# Project Design

Google Audioset contains a total of 5.8 hours of video with manually annotated audio events. Barking dataset is about 0.2 hour long. For this project, the dataset will include barking and other events. Different size of training set will be tested. The smallest dataset will contain barking and 0.2 hour long of other audio events, so it will be balanced. Then the training set will include higher portion of other audio events, to make the dataset larger. Same principle will be followed when gathering dataset for training, validation and testing.

## Preprocessing

Data preprocessing is as follows: Audio from every 10-minute-long YouTube video is extracted. The audio is then being broken down into frames according to the chosen window size. Only audible frequency will be included and others are filtered out. Then each frame is being transformed into time-frequency plots (log-scaled mel-spectrograms). Librosa [8] will be used to create the plots. Each audio input becomes a stack of images with the same label.



Example of a spectrogram

## Training and Evaluation

After preprocessing the data, the different models are going to be trained and evaluated. All models will have an output layer with one node (activation function is sigmoid) to show the probability of the input being a dog bark.

**a. Fully Connected Neural Network**

First a fully connected neural network is fit to be the benchmark model. A grid search will be performed to select the number of layers and the number of neurons in each hidden layer.

**b. CNN from scratch**

Several CNN models would be tried from scratch. The CNNs will at least have two convolutional layers with ReLu activations to pick up relatively simple things like edges and shapes. They will also have two fully connected layers at the end to learn representations.

**c. Transfer Learning**

Transfer learning would be tried attempting to get a higher F-1 score. For transfer learning, only the first few layers are kept because the images in this project’s dataset are not similar to the original dataset. VGG-16 [9] and ResNet50 [10] will be implemented.

# Reference

[1] Hershey, Shawn, et al. "CNN architectures for large-scale audio classification." Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017.
[2] https://www.skcript.com/svr/building-audio-classifier-nueral-network/
[3] https://www.mathworks.com/help/signal/examples/practical-introduction-to-time-frequency-analysis.html
[4] Gemmeke, Jort F., et al. "Audio set: An ontology and human-labeled dataset for audio events." Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on. IEEE, 2017.
[5] Abu-El-Haija, Sami, et al. "Youtube-8m: A large-scale video classification benchmark." arXiv preprint arXiv:1609.08675 (2016).
[6] http://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html#cross-entropy
[7] https://en.wikipedia.org/wiki/F1_score
[8] https://librosa.github.io/librosa/
[9] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).
[10] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.