Project Report – EE698V

ABHAY DAYAL MATHUR

Roll No. 190016

ACOUSTIC EVENT DETECTION

Problem Statement

Task 1

To classify a given sound-clip into one of ten possible classes.

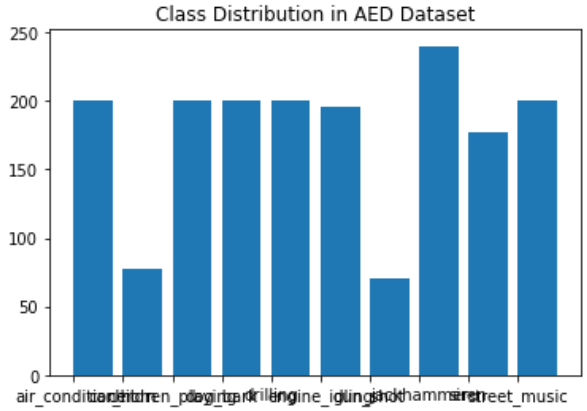
Task 2

To identify the sequence of classes of sounds in a sound clip.

The Dataset

The dataset provided was a collection of 1706 .wav files of varying durations and their corresponding labels (ground truth). The ten classes are – *'air_conditioner', 'car_horn', 'children_playing', 'dog_bark', 'drilling', 'engine_idling', 'gun_shot', 'jackhammer', 'siren'* and *'street_music'*.

Necessary statistics of the dataset can be seen in the figures below.



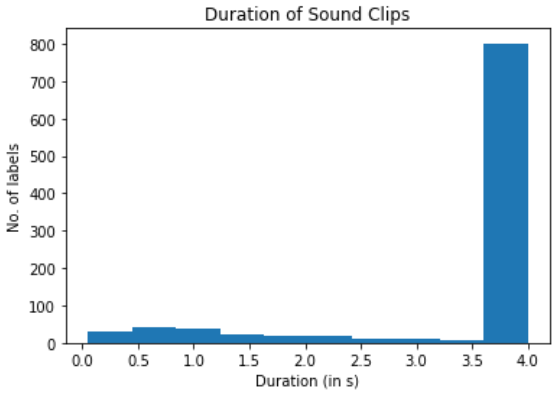


Fig 1. Class Distribution | Fig 2. Histogram for Durations

Data Preparation and Augmentation

The .wav files were converted to interpretable spectrograms using the librosa API (these functionalities were provided in utils.py)

Given the small number of samples in the dataset compared to the large number of trainable parameters the model would inevitably have the samples in the dataset were augmented in order to reduce overfitting in the following two ways [1] –

1. **Frequency Masking**
   Here, *f* consecutive frequency channels in the spectrograms are masked, where the width and location of the band is chosen through normal distributions about defined frequency-parameters.
2. **Time Masking**

Similarly, *t* consecutive time steps are masked, with an upper bound on the mask width.

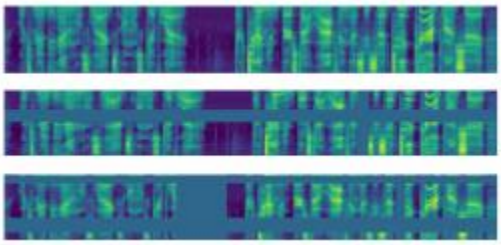


Fig 3. Illustration of Spectrogram Augmentation [1]

The Model

The model used is a *Deep Convolutional Neural Network* which takes padded/cropped spectrograms of 400 timesteps and predicts the class of sound through a *SoftMax* layer. The large input field (of a few seconds) has been taken in accordance with the fact that the event detection problem does not need to model sub-words/n-grams for classification, and a sub-word approach might not encompass the diversity of the sample-space. The network therefore models the entire acoustic event, enabling end-to-end parameter optimization. [2]

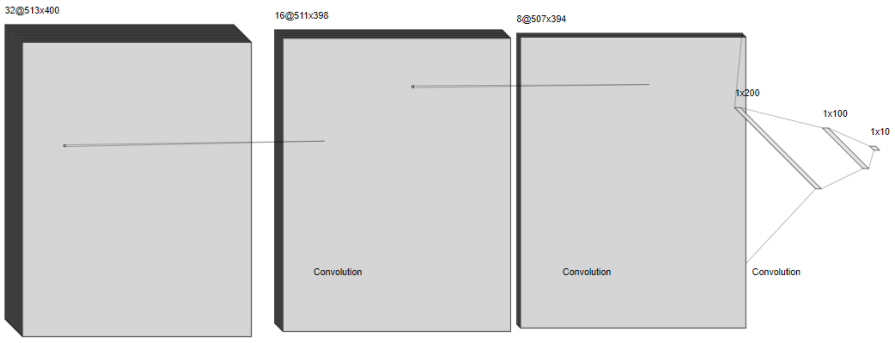


Fig 4. The Model Architecture

Task 1 simply involves feeding the clip to the classification model after padding/cropping and obtaining predictions.

For **Task 2**, the prediction procedure has been defined as follows. To accommodate for events of shorter time durations than those modeled by the network, 4 'beams' are created from the spectrogram of the sound clip by pre-padding and post-padding the spectrogram with (0, 100, 200, 300) and (300, 200, 100, 0) time steps, respectively. *SoftMax* predictions are obtained for each beam and then aggregated to obtain the final predicted 'argmax' sequence. I have refrained from using a sequential model for this task given the independence of consecutives instances in a sample.

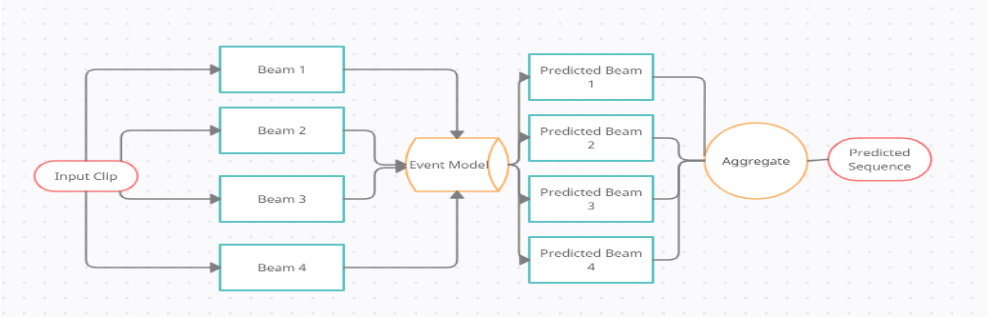


Fig 5. Pipeline for Task 2

Training

Training for the model was carried out in 21 epochs, with the data samples being augmented as explained above after every 7 epochs. The training history can be seen below.

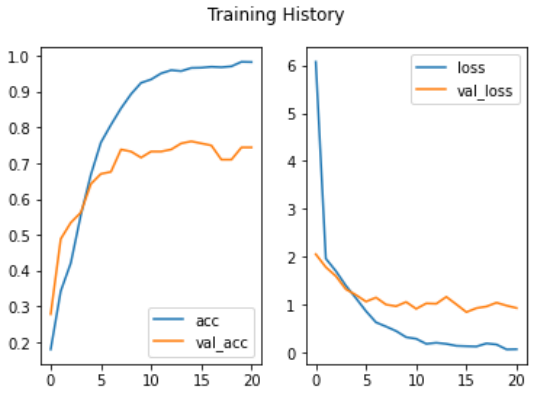


Fig 6. Training History

References

1. Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, Quoc V. Le, “SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition”, *Proc. Interspeech 2019, 2613-2617*
2. Naoya Takahashi, Michael Gygli, Beat Pfister, Luc Van Gool, “Deep Convolutional Neural Networks and Data Augmentation for Acoustic Event Detection”, *Interspeech 2016*