Classifying Exertion Level

Ryan Sheehan, Greg Preston, and Joe Blake

Using audio classification to detect exertion level from speech

- Our app is designed to detect someone's exertion or exhaustion level from their speech
- Specifically short term exhaustion from physical activity, rather than being tired from lack of sleep.

Data Collection

We defined 3 classes for our application to distinguish: ‘High,’ as in high exertion, having recently done something strenuous and therefore having a heart rate above 100 bpm, ‘Low’ as in low exertion, not having done any strenuous activity and a heart rate near resting, and ‘None’ as in no speaker, or ambient noise.

We collected data for our ‘High’ class by having one of our members run to raise their exertion level, and then immediately recording the sample. We collected data for our ‘Low’ class by recording one of our members speaking in a normal, rested state. We collected data for our ‘None’ class by recording audio from a lightly occupied room, with people speaking quietly in the distance.

Issues

Besides the two features that had been implemented in our audio classification assignment, the first feature we thought to use was a rhythmic feature called a ‘tempogram.’ We thought that there may be detectable differences in the rhythmic patterns of a person speaking in an exerted state. Unfortunately using this feature only increased our various accuracy measures by a couple percent.

Fortunately, our next feature was much more impactful. Having had little success with the rhythmic feature we decided to try a pitch based feature, the mel spectrogram. Providing this feature to our classifier increased our accuracy metrics to 90-100%.

Screenshots

Confusion matrix for 1 fold of a 10 fold validation:

```
7  0  0
0 12  0
0  0 11
```

App running:

```
Received audio data of length 8000
Speaker is high.
Received audio data of length 8000
Speaker is high.
```

Results

It seems to work quite well. Despite only being trained on one group member, it was mostly correct when used to classify the other group members as well. Because most of our data was collected in one environment, if it is used in a significantly different environment it tends to be thrown off. It also seems likely that there was a significant amount of background noise when we recorded our ‘High’ samples, as the most common error is mistaking ambient noise for ‘High’.

We were quite happy with the level of accuracy we managed to achieve with only about 90 seconds of data for each class. The logical next step would be to collect more data. If we collected data in more diverse environments and from more speakers, and simply in higher quantity, I think the classifier would be extremely robust.