WildcatOne

Thanks for hosting this challenge. This was my first attempt at working with audio data and I learned quite a bit. I typically work in images, so I may not use appropriate audio terminology. Let me know if you think a more-complete writeup is worthwhile.

AUC 80.7 for our best entry

We implemented in Tensorflow: https://github.com/UkyVision/bird-audio-detection

We tried many different network architectures, but the best result came from network 8.2 (https://github.com/UkyVision/bird-audio-detection/blob/master/src/network.py#L146).

The network:

- raw audio data as input (no spectrograms, just downsample by a factor of 2)
- construct a pyramid by temporal averaging
- extract features from each layer of the pyramid
    - apply a set of convolutional filters (the same weights for all layers)
    - compute the energy
    - normalize the convolutional filters
- concatenate the energy and the normalized convolutional filter activations for all layers of the pyramid
- a few more layers of convolutions

Training:

- ADAM...
- expand short clips by wrapping
- extract a random ~9-second subwindow

Evaluation:

- Average across many random crops for each clip

This network, and many other networks we tried, worked well on held-out training data but never did well on the challenge data. I think that with additional training data these could be made to work quite well on the challenge data.