Data Selection Proposal

Jacky Zhang

20 February 2022

1 Choice of Dataset

The dataset that I am planning to work with is the Cornell Birdcall Identification dataset. I plan to train a model to identify the species of a bird based on audio recordings of its call. I chose this dataset because it is well organized and labeled. It has been used previously as a Kaggle competition, so I can be confident in the quality of the data.

2 Methodology

2.1 Data Preprocessing

Since I will be working with audio files, the first thing I need to do is read and load the audio files, which can be done with `librosa` or similar libraries. Next, I will have to convert the files to have the same number of channels, either mono or stereo. I will also standardize the sampling rate to 44.1 kHz and resize the audio samples to all have the same length by padding it with silence or truncating it. I could then augment the data by using a time shift or by introducing noise. The next step would be to convert the audio to mel spectrograms. I can then augment the spectrogram through frequency and time masking.

2.2 Machine Learning Model

Since I will be working with spectrogram images, I will use a Convolutional Neural Network (CNN) and a linear classifier to predict the bird calls. CNNs usually work well when classifying images, and spectrograms should be no different.

There are two approaches I could take. The first is to train a CNN from scratch and add my own layers. I can look at successful models for inspiration on the architecture of my model. The second is to use a CNN that has been pre-trained on large amounts of data.

Examples of pre-trained models include `VGG16`, `VGG19`, `ResNet`, and `MobileNet`. I can use the pre-trained models for feature extraction by discarding the classification layer. I can also fine-tune the pre-trained models by freezing layers and retraining.

I will try both approaches to see which one performs better.

2.3 Evaluation Metric

Since this is a classification problem, I will be reporting confusion matrix and accuracy, as well as precision and recall through the F_1-score. Looking at papers published on the acoustic detection of birds through deep learning, the best models had an accuracy of around 0.8 to 0.9. On the other hand, the top F_1-score on Kaggle is 0.681 and the baseline is 0.568. However, the Kaggle test data included long audio recordings with multiple bird calls. I plan to set aside 10% of the training set to use as test data, since it makes more sense my application.

3 Application

I will integrate my model into a web app that allows users to upload audio files to be classified. There will also be a button that allows users to record and upload audio using their device's microphone. The output will be an image and label of the predicted bird species. If time permits, I could include addition features such as live prediction, where the model continuously predicts bird calls over short intervals while the user leaves their microphone on. To implement this, I could use `WebSocket` and/or `WebRTC` to stream the audio to the server, or run the model in browser, through `tensorflow.js` for example.