Proposal For COMP90019 Distributed Computing Project – A Bird Species Recognition System Using Audio Data

Kunliang Wu 684226

1 Motivation

Scientific researches often require a lot of samples and raw materials, yet the task of collecting these data is always thought to be a tough one. As the technologies evolve every day, almost everyone in our modern world has a mobile device with him/herself and has easy access to the internet. All of these makes it possible to get every citizen get involved in the data collecting process, which could produce a satisfactory result regarding data collecting while costing much less than the traditional data collecting approaches.

Apart from that, the fast development in Machine Learning and Voice Recognition process has laid a solid foundation for making reasonable predictions without costing too much human resources.

This is why we would like to try taking advantages of both two main strategies and techniques and building a bird species recognition system using audio data, while getting a good model for latter study as a valuable by-product.

2 Overview

2.1 Background of the project

There have been many study about the feature engineering and machine learning algorithms for species recognition. Some transform the voice data collected to vectors in regard to minimum frequency, maximum frequency, maximum power and call duration. In other study, a feed-forward artificial neural network is used to categorize a pre-defined bird species set, which also achieved satisfying results. However, the most prevalent and predominant approach used is Mel-frequency cepstral coefficients (MFCC). This strategy uses a cosine transform to generate a vector from the audio data.

As for data collecting and storing, a server set up on a Cloud Service Provider with a non-SQL database is a good option for the project. The logics for background analysis would be handled by some mature libraries from third parties.

2.2 Abstract description of the project

Server uses the information collected from the mobile clients and other audio dataset for birds, like http://www.xeno-canto.org/ in order to generate a well-developed model for bird species recognition. The classified results with good confidence will then put into model improvement process later on in return. As the size of the database grows, the model will gradually become stable and more appropriate for future prediction. The prediction results for bird species would be returned to the client users.

3 Features

- Client terminals for multiple platforms :

  The project will involve client software on three different platforms – Android, iOS and Web-browser.

- Left unsatisfying prediction results to the experts:

  For those results the classifiers are not so confident with, they will then be kept for further inspection and classified by experts on birds.

- Geographic location with the recording place and other information (features) for bird's voice.

  The system will demonstrate the location where the audio file is recorded and other significant from the users which might be helpful to build a better model.

  Also, some relevant briefings about the birds' kinds will be provided to the user. Like a picture or habitat description of it for users to make further judgement of the results.

- Renew the model when enough new and confident results are collected

  The system will choose a hurdle for the number of new audio records added. When the hurdle is reached, the system will then trigger a renew process trying to generate a model with better performance.

- Flexible Model Module:

  To get an acceptable results of the prediction, different pre-processing strategies and algorithms would be applied to the system, while only the one with the best performance would be exploited to the users. This means that if the researchers can come up with a method to build a better model, they can reuse the existing system with minimum coding to replace the current model module in use.

4 Methodology

4.1 Design Phase

In this phase, the project will be broken down into several models. Generally speaking, these models are: Communication Module, Pre-processing Module, Machine Learning Module, Database Module and Web Demonstration Module. During the design phase, the focus at this stage would be the protocols and interfaces design. When the interfaces are completely settle, we can look into the individual module separately and finish the data-stream logic design.

4.2 Implementation Phase

The major tasks for this phase is to finish the coding for different modules. This requires the developer to have a good understandings of what libraries to use and the pros and cons that go with them. Some pre-study needs to be done here for a good libraries selection. After the coding part is over, we shall focus on the server setting up using Cloud Service Provider.

4.3 Testing Phase

After the entire project is completely built, the next step would be testing how the system perform and how the tunings can be done to achieve a better results. Bugs are ought to be discovered and then removed here as well.

For the evaluation, the system needs to handle the voice data the users submit in a variety of qualities. The performances of the system in different environments should also be taken into consideration.

5 Potential Risks and Problems

5.1 Limited Training Resources

Currently I only found four major sites offering audio data files which can be used for training. They are:

1.1 http://www.bioacoustica.org/gallery/aves_eng.html

1.2 http://soundbible.com/tags-bird.html

1.3 http://www.graemechapman.com.au/resources/recording-bird-sound.htm

1.4 http://www.xeno-canto.org/

Even though they provide the audio files free for use, but they don’t have any bulk access for all the dataset, and that means we have to build a tool to grab the raw materials from these sites by ourselves. Furthermore, these dataset is relatively small compared to other similar research studies – averagely it requires 1000 records to capture the features to give a confident prediction. This problem is also related to 5.2 Quality Issues and 5.5 Huge Collection for Species to Be Classified.

5.2 Quality Issues

In Xeno-Canto, the audio data submitted by the users are grouped according to the voice quality. Here are a table for the records distribution without counting Quality E in the set:

Records in AU/Total	Quality A	Quality B	Quality C	Quality D
4190 results from 561 species	2318 results from 413 species	1324 results from 431 species	441 results from 270 species	36 results from 35 species
316952 results from 9619 species	110016 results from 8013 species	123242 results from 8642 species	53500 results from 7081 species	11698 results from 3275 species

Table 5.2 - The distribution of the voice records

My current idea is use the files in Quality A for a prototype and check the possible improvements taking Quality B and C into training dataset.

For the users input, the temporary hurdle for the audio submitted is Quality C. This is to say that the first version of the system won’t accept any input lower than Quality C.

5.3 Noise Control

I am still quite confused about the noise control strategy at the moment – since this may have some impacts on the pre-processing and feature engineering stage. In other words, whether to leave the task for noise control to Clients or to the server is a controversial problem. If both clients and servers have the same logics dealing with noise, then it won’t be necessary to do it twice on both ends. Another challenges are how to get the main sound with other sounds interfere, and how to define the criteria for “pre-processing” to get the audio files to meet the standards which are used in training, developing and testing.

5.4 Training Strategy

There are two main popular strategies - feed-forward artificial neural network and MFCC. The first one allows an ever-evolving model which can adjust itself as more and more data is given yet it requires a relatively large start-up training set and performing well with limited numbers of possible result tags, while the second one needs to rebuild the model every time when there comes more records. My idea is that the first step is to finish a server using MFCC and then trying the other one when there is enough time.

5.5 Huge Collection for Species to Be Classified

As it can be seen from table 5.2, there exist two main approaches for, we only have limited records about the birds in Australia. A simple solution is to tailor the size of the bird species to be classified, since that a multiple classifiers are much harder than a positive/negative classifier. For the prototype, I think it would be suitable to restrict the species number to the top 10 birds which can be widely seen across Australia, and adding more to the collection as the project moves forward.

6 Hardware and Software Requirements

I will apply for some Nectar resources which allows the team to get a relatively fast modelling speed. The requirements would be submit in the middle of August. As for the software requirements, the team will discuss it in details and add more about the libraries in the following discussion and meeting sessions.