CLASSIFICATION OF ANIMAL SPECIES USING SOUND RECOGNITION

Vinu Pradeep Menon (0731928) *Ashwin Albert (0731914)*

Jacob K Varghese (0729204)

5th April 2020

ABSTRACT

Around the world as the technological development has been very much advancing, audio processing technology has also been reaching to its advanced level of improved quality and operational establishment. As per our findings there are many kinds of researches so far mainly focused on speech which bought to an epic speech recognition system which we all have in our user-friendly electronic equipment's. Whereas very few of researches that did focused on the animal voice recognition technology that could help the animal species who are forced to change their own living places and their natural ways of living. And the reason behind this are the other kinds of activities which creates pollution. The presence of animals which are living in bare places are hard to identify in a physical manner and if identified it would be adamantine to invigilate them. But if we do not take effort to monitor them through out those species could experience dangerous circumstances which we already now cases that had happened around the world and the most in places such as Thailand, China, Uganda, Tanzania, Zimbabwe, Kenya and Philippines. Audio Recognition is one of the best and effective ways of tracking those certain species by their voices. But classifying sound can be a challenging task. Audio signals and their pre- processing techniques stimulate to track the audio from these species. There are various ways of recognizing audio. Training machines by audio files which are already been recorded, or by recording them live and distinguishing them. Audio signals of the species can be encoded by suppressing all the background noise and echoes. The detection of various signals in voice is the method we concentrate on, which in terms of machine learning is known as audio recognition.

Machine Learning algorithm neural network were used to train the machines with the data sets of previously identified important species. Neural Network is one of the most popularly used machine learning algorithm, its accuracy and speed is ample for this project. Deep learning techniques such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are used in this project. This document primarily serves as a technical paper explaining how we have applied such programme to meet the adequate result.

INTRODUCTION

Safeguard of animals and birds that live in the forest need to be certified. Most of the species on land exist in the forests, which implies the value of preserving the balance of nature in terms it is called equilibrium of eco-system. It is mostly important to understand how human demand on nature to attain the economy and to hold up livelihood is affecting the animals and their habitat. This approach is usable for identifying animals who could be affected by human unnatural behaviour and activities. And, this method could be further improved and used to get visual of species which have not yet been discovered, those which may be in the dense forest where human existence is dangerous or not possible.

The only way to track those species which are in deep forests with no end for trees is through audio and video or image. It is in fact difficult to keep track on these species. Whereas if we use a tool that recognises signals form voices of different species, it could be used cover tracking a wide range with minimal devices. This will save the government a lot of money foe their approach to save animals or birds while they are in an emergency state.

Datasets with and without background noise are used, in the future we can also take an approach to use background noise and implement different methods to save the environment by recording specific background noise from forest fires, with machinery cutting down trees. We can also take these into account as a safeguard measure to protect the environment.

The dataset we have used is created by combining voices of species obtained from various open sources available on the internet. It contains sound file in wav format of minimum of 10 different species which are bear, wolf, snake, seal, panther, leopard, coyote, jaguar, cougar and cheetah. In terms of analytics we have more than 77 voice samples with 10 different classes. The csv file consists of 2 features namely File name subsume the name of each sound files and Species Type subsume the corresponding class for each sound file. The sound files are the independent variable. The Species Type will be the target variable. It is to be prominent that the feature File name will be only used to map the independent variable to the target variable. The analytical tools which we have used so far are Excel, Anaconda Navigator, Python, Jupyter, Notebook, Media Coder. The python packages and modules we have used are librosa (audio processing package), Keras (high level neural network), labelEncoder (label creation) and TensorFlow (python library used to create deep learning models).

RELATED WORKS

Critical study of similar research in the area of audio recognition using machine learning

- Article - ‘Animal Sound Classification Using A Convolutional Neural Network’
  Authors - Emre Sasmaz and F. Boray Tek (Dept of computer engineering, Turkey)

The project includes animal sound classification using deep learning and has proposed a method based on the convolutional neural network architecture. As inputs to the network, sound files were pre-processed to extract Mel Frequency Cepstral Coefficients (MFCC) by using the Librosa library which we both have done in common. In order to train and check the program they obtained 875 animal sound samples from an online source site for 10 different animal types. Done with Classification confusion matrix, the output is obtained by different gradient descent optimizers. The highest accuracy of 75% attained with Nesterov - accelerated Adaptive Moment Estimation (ADAM) which is an optimization algorithm for stochastic deep learning models.

Content: -
https://ieeexplore.ieee.org/document/8566449

- Article - 'Animal Detection using Deep Learning Networks'
  Authors - Rashmi Jayakumar, Sanchithaa Harikumar, N Banupriya, S Saranya
  (Ramakrishna Engineering College, India)
  The work possesses on monitoring of wild animals in their habitat. They have created an algorithm that could detect animal presence in the forest. The algorithm they have created classifies animals based on the specified animal's image. They are also concentrating on preventing animal-vehicle accidents, tracing lost animals and animal thefts. Deep learning algorithm Convolutional Neural Network was used in order to attain the result. Apart from what we have done, they have used a template matching algorithm which is basically a method of finding similar parts of image from a template image. They were able to classify animals efficiently with accuracy more than 79% for each image. They intend to improve their project by adding animal detection notification in the form of an alert text message.

Content:-
https://www.researchgate.net/publication/335813356_Animal_Detection_Using_Deep_Learning_Algorithm

- Article - 'Voice Classification with neural network'
  Authors – Jurgen Arias

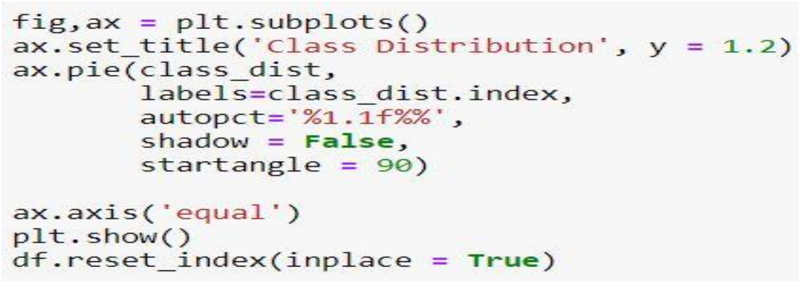
The problem the work focused on is to classify the sound of different urban noises like playing children, Street music, car sound etc. The algorithm used is Neural Network (NN) and convolutional neural network (CNN). The first step done is extracting numerical features from the sound data with the help of librosa library and then train a NN model and the accuracy acquired is 93% on test data. The second step is converting the sound data to images and then used those images to train convolutional neural network and got accuracy of 92% on test data.

Content: -
https://towardsdatascience.com/voice-classification-with-neural-networks-ff90f94358ec

METHODS

1. Data preparation:
   The data where collected from various sources and each and every samples were converted to wav format using audio converter application. The collected 77 samples of 10 different species where named accordingly and was saved under one folder named wavfiles. Also, a csv file corresponding to the wav files was created which contained the file name as well as their corresponding class. Both these files, wav files as well as the csv files, where then inputted into the python notebook for further processing.

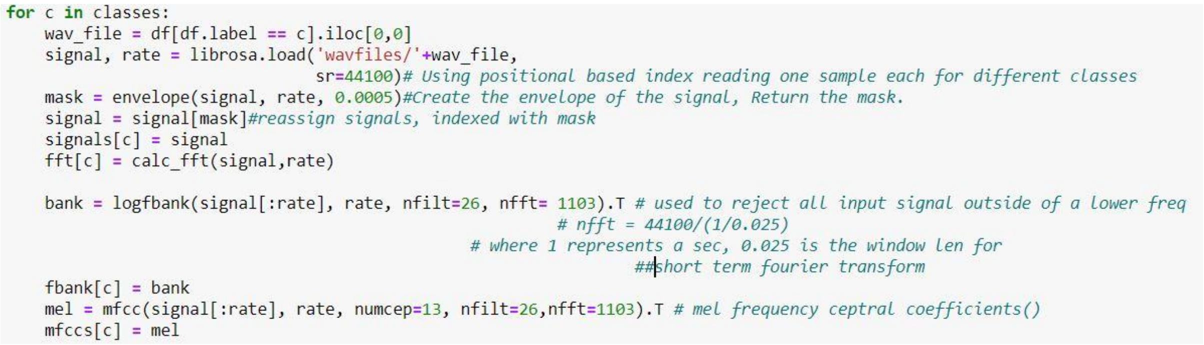
2. Data Exploration:
   The distribution of various classes were determined using a pie chart.



```
fig,ax = plt.subplots()
ax.set_title('Class Distribution', y = 1.2)
ax.pie(class_dist,
        labels=class_dist.index,
        autopct='%1.1f%%',
        shadow = False,
        startangle = 90)

ax.axis('equal')
plt.show()
df.reset_index(inplace = True)
```

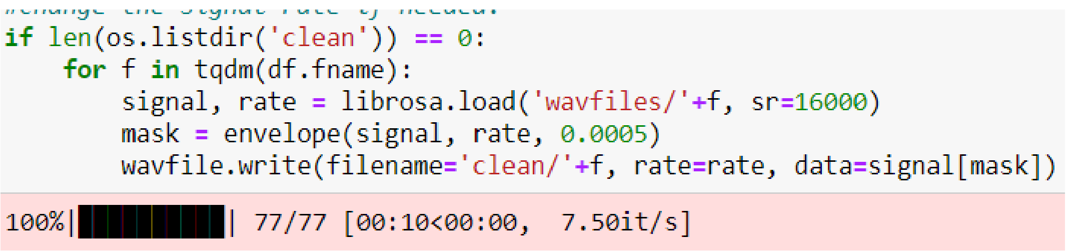
3. Data preprocessing:
   The audio files where were converted into various other formats such as time series format, fast Fourier transform(FFT), Filter bank and Mel-frequency Ceptral Coefficient(MFCC) for further understanding of each wav file. The following represents the process to calculate filter bank coefficients and MFCC.



```
for c in classes:
    wav_file = df[df.label == c].iloc[0,0]
    signal, rate = librosa.load('wavfiles/'+wav_file,
                                sr=44100)# Using positional based index reading one sample each for different classes
    mask = envelope(signal, rate, 0.0005)#Create the envelope of the signal, Return the mask.
    signal = signal[mask]#reassign signals, indexed with mask
    signals[c] = signal
    fft[c] = calc_fft(signal,rate)

    bank = logfbank(signal[:rate], rate, nfilt=26, nfft= 1103).T # used to reject all input signal outside of a lower freq
                                                     # nfft = 44100/(1/0.025)
                                            # where 1 represents a sec, 0.025 is the window len for
                                                          ##short term fourier transform
    fbank[c] = bank
    mel = mfcc(signal[:rate], rate, numcep=13, nfilt=26,nfft=1103).T # mel frequency ceptral coefficients()
    mfccs[c] = mel
```

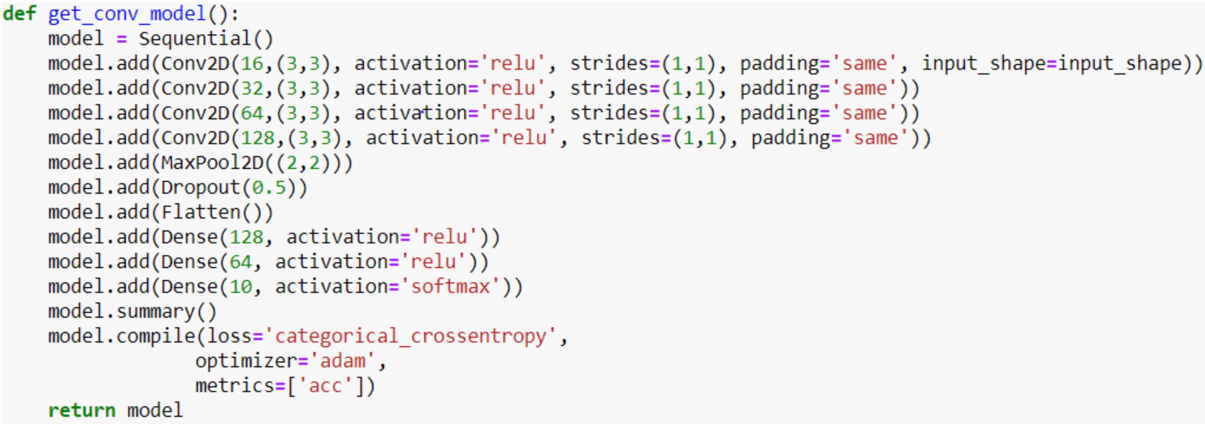
The audio files are then down sampled with a sample rate of 16000 Hertz and the cleaned files are saved to a new folder named clean. From now on these files will be used for the modelling. The following is the process of down sampling the audio files.



```
#change the signal rate if needed.
if len(os.listdir('clean')) == 0:
    for f in tqdm(df.fname):
        signal, rate = librosa.load('wavfiles/'+f, sr=16000)
        mask = envelope(signal, rate, 0.0005)
        wavfile.write(filename='clean/'+f, rate=rate, data=signal[mask])

100%|██████████| 77/77 [00:10<00:00,  7.50it/s]
```

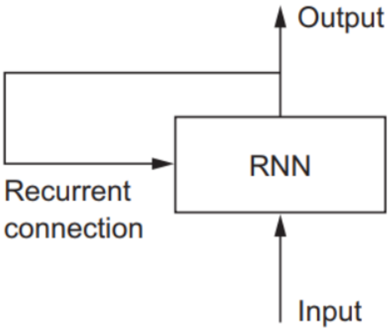
4. Modelling:
   CNN (Convolutional Neural Network): This a deep learning neural network model basically used for image processing and image classification. Since the audio files that are converted to the mfcc format is used for building this model. The following steps represents the building of CNN.



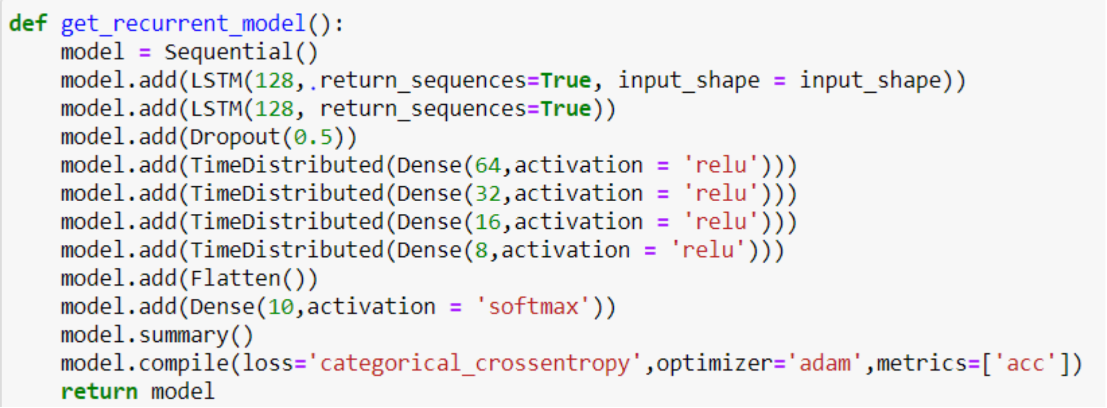
```
def get_conv_model():
    model = Sequential()
    model.add(Conv2D(16,(3,3), activation='relu', strides=(1,1), padding='same', input_shape=input_shape))
    model.add(Conv2D(32,(3,3), activation='relu', strides=(1,1), padding='same'))
    model.add(Conv2D(64,(3,3), activation='relu', strides=(1,1), padding='same'))
    model.add(Conv2D(128,(3,3), activation='relu', strides=(1,1), padding='same'))
    model.add(MaxPool2D((2,2)))
    model.add(Dropout(0.5))
    model.add(Flatten())
    model.add(Dense(128, activation='relu'))
    model.add(Dense(64, activation='relu'))
    model.add(Dense(10, activation='softmax'))
    model.summary()
    model.compile(loss='categorical_crossentropy',
                  optimizer='adam',
                  metrics=['acc'])
    return model
```

The activation function used here is relu which stands for Rectified linear unit. It is defined as f(x) = max(0,x). That is if the input value is positive it takes that same value or else it takes 0. The loss function used here is categorical cross entropy. And the metric used is acc (Accuracy).

RNN(Recurrent Neural Network): Unlike CNN, RNN uses memory blocks instead of neurons. This memory blocks uses the output of the previous layer to remember its preceding elements. LSTM is on of the main feature of RNN. The following is the basic diagram of RNN.



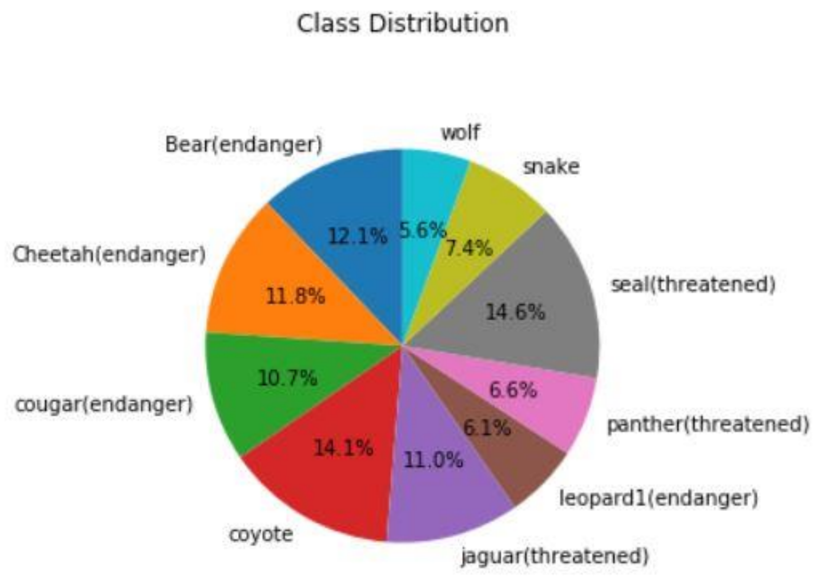
The following represents the model definition.



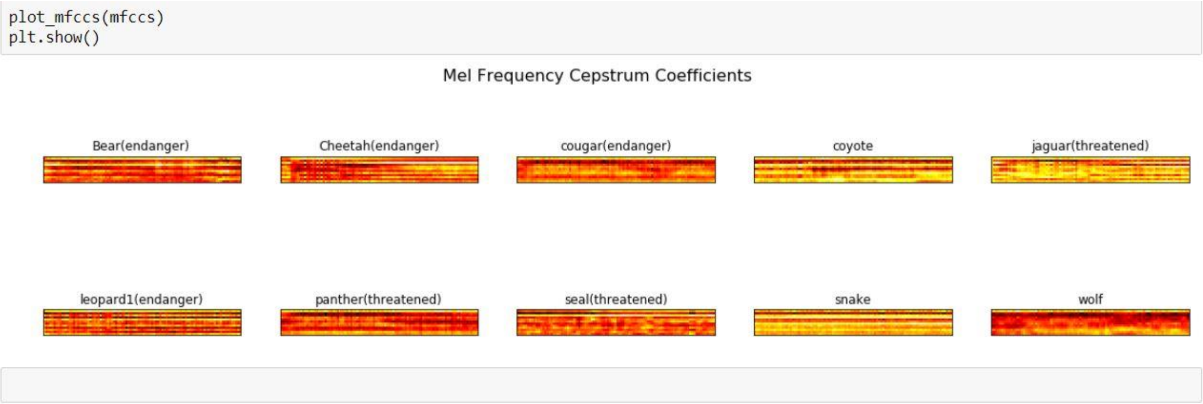
```
def get_recurrent_model():
    model = Sequential()
    model.add(LSTM(128,.return_sequences=True, input_shape = input_shape))
    model.add(LSTM(128, return_sequences=True))
    model.add(Dropout(0.5))
    model.add(TimeDistributed(Dense(64,activation = 'relu')))
    model.add(TimeDistributed(Dense(32,activation = 'relu')))
    model.add(TimeDistributed(Dense(16,activation = 'relu')))
    model.add(TimeDistributed(Dense(8,activation = 'relu')))
    model.add(Flatten())
    model.add(Dense(10,activation = 'softmax'))
    model.summary()
    model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['acc'])
    return model
```

RESULTS

Plotting the class distribution of each species



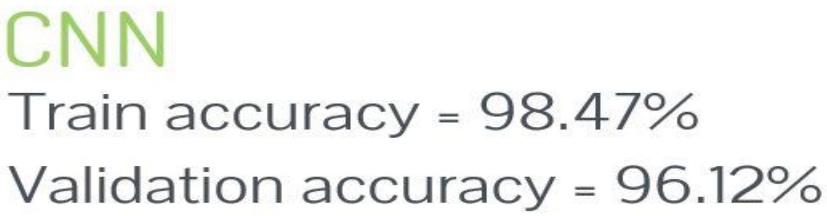
Mel frequency cepstrum coefficients



```
plot_mfccs(mfccs)
plt.show()
```

Results from the models

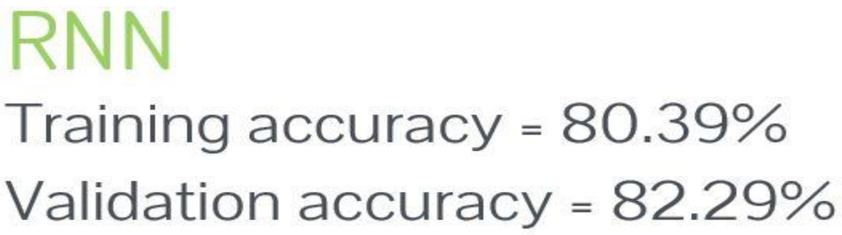
The number of epochs used for each model was 20.



CNN

Train accuracy = 98.47%

Validation accuracy = 96.12%



RNN

Training accuracy = 80.39%

Validation accuracy = 82.29%

DISCUSSIONS

Using two different models such as convolution neural network (CNN) and Recurrent neural network (RNN) we have achieved a result. The training and validation accuracy of CNN model was way better than RNN. The result read with a training accuracy of 98.47% and validation accuracy of 96.12% for the CNN model whereas, it is 80.39% and 82.29% respectively for the RNN model. The results are satisfying because the machine can achieve more than 90% accuracy in one of the models.

One of the setbacks that we have faced during the model building was the lack of enough data sources During the down-sampling of the audio where the audio files are cleaned and generated, we lost a portion of our data due to low quality as a result there were not enough data available for the testing. since we do not have enough data for testing, we then tried to do find out the validation accuracy and we were able to achieve it with the same set of datasets. The model can be used to predict different classes of animals once new data is available. Collecting more samples for each class would help to test the accuracy of the model. In future when enough data available it would easy do the train the model and successfully test the model accuracy.

CONCLUSION

This project was started with a very simple idea of classifying endangered animal species using deep learning techniques like the Artificial Neural Networks. But, as the project progressed, we came to know that sound classification and collecting the require data for the same is a tough task. Hence, we took a lot of time learning about each model before starting the project.

The collection of data was the key to building a successful model. The lack of good quality data was on of the main factor that pulled down the success of the project. But on the other hand, with the available data we were able to build 2 working models (RNN and CNN).

By comparing the validation accuracy we were able to conclude that CNN was the better model when compared to RNN which was not up to the mark. Although CNN model showed a validation accuracy of 96.12% the reliability of the model still remains an issue due to the quantity of the data. But if provided with good quality of data the models can be retrained to get an even better and reliable result.

CONTRIBUTIONS

Ashwin Albert

- Data Preparation
- Data pre-processing techniques
- Plotting and Cleaning
- Research on related works
- Report Making

Jacob Kalloor Varghese

- Data collection
- CSV file creation
- Model building
- Research on related works
- Report Making

Vinu Pradeep Menon

- Data pre-processing
- Data preparation
- Model Building
- Report Making

REFERENCES

https://anaconda.org/
https://en.wikipedia.org/wiki/List_of_animal_sounds
http://soundbible.com/tags-animal.html
https://github.com/libphy/which_animal
https://glassboxmedicine.com/2019/02/17/measuring-performance-the-confusion-matrix/
https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9
https://medium.com/@mikesmales/sound-classification-using-deep-learning-8bc2aa1990b7
https://towardsdatascience.com/recurrent-neural-networks-by-example-in-python-ffd204f99470
https://www.youtube.com/watch?v=Z7YM-HAz-IY&list=PLhA3b2k8R3t2Ng1WW_7MiXeh1pfQJQi_P&index=1

APPENDICES

Data preparation and pre-processing: EDA.ipynb

Modelling: Models.ipynb