Written Homework 3 : Deep Learning

Title : Exploring Neural Networks for identifying the sounds of birds common in the Seattle area.

Abstract

In this study, the application of neural networks for the classification of bird species based on their calls is explored, with a focus on the avian population of the Seattle area. Spectrograms derived from sound clips obtained from the Xeno-Canto [1] bird sound archive are preprocessed, and custom neural network architectures are constructed for binary and multi-class classification tasks. For the binary classification task between the American Crow (Amecro) and Blue Jay (BlueJay) birds, an initial model with three convolutional layers using ReLU activation function yielded an accuracy of 48%. Due to the low accuracy, parameter tuning including batch size reduction was performed. Subsequently, several models were built and validated, resulting in a final model with an improved accuracy of 96%. Plots depicting epoch versus loss and epoch versus accuracy were examined to assess model performance. For the multi-class classification task involving all twelve bird species, various models with two convolutional layers were developed, achieving accuracies ranging from 63% to 71%. By incorporating three convolutional layers, the accuracy was boosted to 72.38%. Confusion matrices were analyzed to evaluate the predictions made by this model. Furthermore, the trained model was applied to predict the species present in the given test dataset of three test sounds, classifying them as the bird call of a Dark-eyed Junco. The findings of this study contribute to the understanding of utilizing machine learning for avian species identification and provide insights into the performance of neural network architectures in bioacoustic research.

Introduction and Overview:

Birdsong identification is fundamental in the ornithological world because it helps in conservation efforts and ecological research. Machine learning especially neural networks have made automatic bird species categorization based on sound an area of great interest. This research project therefore examines how neural networks can be used to identify bird songs around Seattle using various species available.

The dataset utilized in this study is sourced from the Xeno-Canto bird sound archive, comprising spectrograms derived from wild sound clips. These spectrograms encapsulate both temporal and frequency information, providing a comprehensive representation of bird vocalizations essential for species identification.

The project is structured around several key objectives. Initially, a binary classification model is developed to differentiate between the calls of the American Crow (Amecro) and the Blue Jay (BlueJay) birds. Through iterative refinement and parameter tuning, the aim is to achieve high accuracy in this binary classification task. Following this, the scope is expanded to classify all twelve bird species present in the dataset. Custom neural network architectures are designed and experimented with to achieve accurate classification across multiple species.

Throughout the project, comprehensive evaluations of the developed models using accuracies are conducted. To learn more about model behavior and performance patterns, visualizations of the model training process such as plots showing epoch vs accuracy and epoch versus loss are also carefully examined. Additionally, the trained models are applied to predict the species present in unlabeled test clips, serving as a validation of their generalization capability to new and unseen data. This project intends to enhance automated machine learning algorithms for bird species identification through rigorous testing and analysis.

Theoretical Background:

Classification:

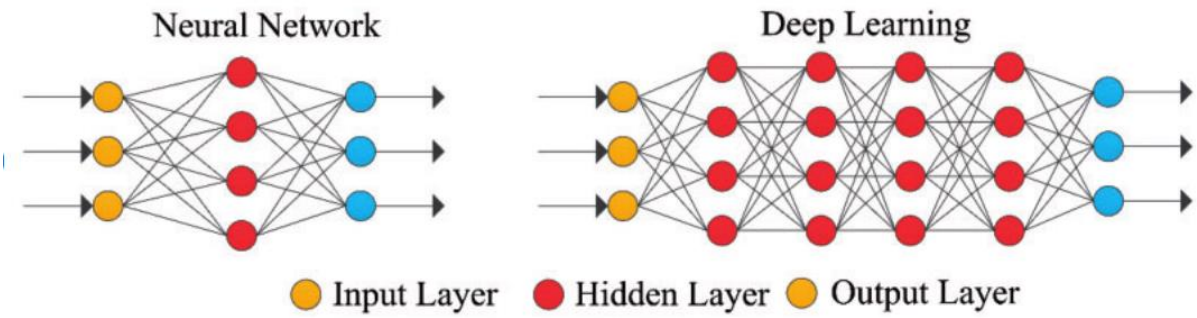
In artificial intelligence and machine learning, classification refers to the machine's ability to assign the instances to their correct groups. This is done through learning a function that maps input features to a discrete target variable, with the target variable being predefined classes or categories.

Deep Learning and Neural Networks:

Deep learning is a subset of machine learning that utilizes neural networks with multiple layers to automatically learn intricate patterns from vast amounts of data. It is powerful in various fields, including computer vision, natural language processing, and speech recognition, enabling unprecedented advancements in tasks such as image classification, language translation, and autonomous driving.

Neural networks are computational models inspired by the human brain's structure and function. They consist of interconnected layers of neurons, each performing simple computations. Neural networks are capable of learning from data through a process known as training, where they adjust their internal parameters to minimize the difference between predicted and actual outputs.

[2]



Deep learning diagram.

Components of Neural Network:

1. **Input Layer:**

   The input layer receives the raw data and passes it on to the next layer.

2. **Convolutional Layers:**

   Convolutional layers (Conv2D) apply filters to the input data, extracting features such as edges, shapes, and textures. These layers are essential for processing spatial information in tasks like image recognition.

3. **Pooling Layers:**

   Pooling layers (MaxPooling2D) reduce the spatial dimensions of the feature maps generated by convolutional layers, helping to decrease computational complexity and focus on the most important features.

4. **Flattening Layer:**

   The flattening layer (Flatten) converts the multi-dimensional feature maps into a one-dimensional vector, preparing them for input into the fully connected layers.

5. **Fully Connected Layer:**

   Fully connected layers (Dense) process the flattened feature vectors and perform complex computations to generate the final output.

Hyper Parameters of Neural Network:

1. **Filters:**

   The number of filters in convolutional layers determines the complexity and capacity of the model to capture features.

2. **Kernel Size:**

   The kernel size specifies the dimensions of the convolutional filters.

3. **Activation Functions:**

   Activation functions introduce non-linearity into the neural network, allowing it to learn complex relationships in the data.

4. **Input Shape:**

   The input shape defines the dimensions of the input data.

5. **Optimizer:**

The optimizer determines how the model's weights are updated during training to minimize the loss function.

6. **Loss Function:**

   The loss function measures the difference between the predicted output of the model and the true labels.

7. **Epochs:**
   An epoch is one complete pass through the entire training dataset.

8. **Batch Size:**
   The batch size determines the number of samples processed before updating the model's weights. It affects the speed and stability of the training process.

9. **Validation Split:**
   This parameter splits the training data into training and validation subsets. If validation_split=0.2 is given, it reserves 20% of the training data for validation during training, allowing the model to monitor its performance on unseen data and detect overfitting.

10. **Metrics:**
    Metrics are just function arguments to save some useful information of the model like measures used to evaluate during training. It does not affect the model directly

Parameter Tuning:

When parameters are tuned for Neural network models, adjustments are made to maximize the model's effectiveness. It's similar to tuning a musical instrument to achieve the best possible sound. By tweaking settings such as the Neural network hyper parameters like epoch, batch_size and number of layers etc., the accuracy and applicability of the model can be enhanced. Parameter tuning ensures that the model performs at its best for the specific dataset and issue at hand.

Methodology

Data clean up or pre-processing:
The initial dataset was obtained from Xeno-Canto, a crowd-sourced bird sounds archive, comprising recordings of a wide variety of bird species. However, for this study, the focus was narrowed to only 12 selected species: the American Crow, Barn Swallow, Black-capped Chickadee, Blue Jay, Dark-eyed Junco, House Finch, Mallard, Northern Flicker, Red-winged Blackbird, Stellar's Jay, Western Meadowlark, and White-crowned Sparrow. These species' clips were chosen for further analysis, with the remaining bird sounds excluded from consideration. Data preprocessing involved several steps: subsampling the sound clips to a 22050 Hz sample rate, identifying "loud" segments exceeding 0.5 seconds, selecting 2-second windows containing bird calls, and

generating spectrograms for each window, resulting in a 343 (time) x 256 (frequency) representation. Each species' bird calls were saved individually, leading to variations in sample counts across species.

For binary classification, the focus was narrowed to the distinction between the American Crow and the Blue Jay. To ensure balanced datasets, the top 50 samples from each of these species were selected. For multiclass classification, the top 35 samples from each of these 12 species were selected to have a balanced dataset. This was because the lowest number of samples was 36 for Mallard and Western Meadowlark.

Data Visualisation:

After the preprocessed dataset was loaded, a sample spectrogram of each bird species was plotted and visualized. This provided insights into the patterns of each species and aided in establishing connections between different calls.

Train and Test Data set Splitting:

For all the models, the data was divided into a training set of 75% and test set of 25%. Further, the validation set was 20% of training data for the models.

Models:

In this analysis, there were various models built for Binary and Multi class Classification. Few of the models are as below.

Binary classification models :

There were 3 models build for distinguishing between the American Crow (Amecro) and the Blue Jay (BlueJay).

Model 1 comprises of three convolutional layers: the first layer comprises 32 filters with a 3x3 kernel and Rectified Linear Unit (ReLU) activation function, followed by a second layer with 64 filters and a 3x3 kernel with ReLU activation, and finally, a third layer with 128 filters and a 3x3 kernel with ReLU activation. Each convolutional layer is followed by a max-pooling layer with a size of 2x2. Subsequently, the model contains a dense layer with 128 units and ReLU activation, followed by a single unit with a sigmoid activation function. Prior to the first dense layer, a dropout layer with a dropout rate of 0.5 is implemented to reduce overfitting. The model is compiled using the RMSprop optimizer and binary crossentropy loss function, with accuracy chosen as the evaluation metric. During training, the model is trained for 20 epochs with a batch size of 75 samples. Upon evaluation, the model achieves an accuracy of 92.00%.

Model 2 comprises two convolutional layers. The first convolutional layer utilizes 32 filters with a 3x3 kernel and Rectified Linear Unit (ReLU) activation function, followed by a second layer with 64 filters and a 3x3 kernel with ReLU activation. Each convolutional layer is succeeded by a max-pooling layer with a size of 2x2. Subsequently, the model incorporates a dense layer with 128 units and ReLU activation, followed by a single unit with a sigmoid activation function for binary classification. The model is optimized using the RMSprop optimizer and employs binary crossentropy as its loss function. The primary evaluation metric used is accuracy. During training, the model undergoes 10

epochs with a batch size of 20 samples. Upon evaluation, the model achieves an accuracy of 96.00%.

Model 3 is constructed with three convolutional layers. The first convolutional layer employs 32 filters with a 3x3 kernel and Rectified Linear Unit (ReLU) activation function, followed by a second layer with 64 filters and a 3x3 kernel with ReLU activation, and finally, a third layer with 128 filters and a 3x3 kernel with ReLU activation. Each convolutional layer is followed by a max-pooling layer with a size of 2x2. Subsequently, the model integrates a dense layer with 128 units and ReLU activation, followed by a single unit with a sigmoid activation function for binary classification. The model is optimized using the Adam optimizer and employs binary crossentropy as its loss function. Accuracy is utilized as the primary evaluation metric. During training, the model is trained for 10 epochs with a batch size of 32 samples. Upon evaluation, the model attains an accuracy of 96.00%.

Multi Class classification models :

There were many models constructed for classifying the 12 different species. Below are 4 noted models.

Model 1 is constructed with two convolutional layers. The first convolutional layer utilizes 32 filters with a 3x3 kernel and Rectified Linear Unit (ReLU) activation function, followed by a second layer with 64 filters and a 3x3 kernel with ReLU activation. Each convolutional layer is followed by a max-pooling layer with a size of 2x2. Subsequently, the model incorporates a dense layer with 128 units and ReLU activation, followed by a dense layer with 12 units and a softmax activation function for multi-class classification. The model is optimized using the Adam optimizer and employs sparse categorical crossentropy as its loss function. Accuracy is utilized as the primary evaluation metric. During training, the model undergoes 10 epochs with a batch size of 10 samples. Upon evaluation, the model attains an accuracy of 69.52%.

Model 2 mirrors the architecture of Model 1, with the exception of the last dense layer, which comprises 12 units and a softmax activation. Upon evaluation, the model achieves an accuracy of 63.81%.

In Model 3, modifications were introduced compared to the previous model. Firstly, the number of filters in the convolutional layers was reduced, with the first convolutional layer utilizing 16 filters and the second layer using 32 filters, both employing a 3x3 kernel and Rectified Linear Unit (ReLU) activation function. Additionally, a dropout layer with a dropout rate of 0.5 was introduced before the first dense layer to alleviate overfitting. Moreover, the batch size during training was doubled from the previous model. These adjustments aimed to enhance the model's generalization capability and improve training efficiency. Despite the reduced complexity, Model 3 achieved a commendable accuracy of 71.43%.

In Model 4, a third convolutional layer was introduced to enhance the model's capacity for feature extraction. The architecture consists of three convolutional layers: the first layer utilizes 32 filters with a 3x3 kernel and Rectified Linear Unit (ReLU) activation function, followed by a second layer with 64 filters and a third layer with 128 filters, all employing ReLU activation. Each convolutional layer is followed by a max-pooling layer

with a size of 2x2. Subsequently, the model integrates three dense layers: the first dense layer contains 256 units with ReLU activation, the second dense layer contains 128 units with ReLU activation, and the final dense layer contains 12 units with a softmax activation function for multi-class classification. Additionally, a dropout layer with a dropout rate of 0.5 was introduced after the first dense layer to mitigate overfitting. The model is optimized using the Adam optimizer and employs sparse categorical crossentropy as its loss function, with accuracy utilized as the primary evaluation metric. During training, the model undergoes 10 epochs with a batch size of 15 samples. Upon evaluation, Model 4 achieves an accuracy of 72.38%.

External Test Data classification :

The labels for three external test datasets were predicted using the Model 4 of the above mentioned multi-class classification. However, the provided external test data were in the mp3 format and varied in length. So, the preprocessing steps as discussed earlier had to be applied to the test data files before utilizing them for prediction.

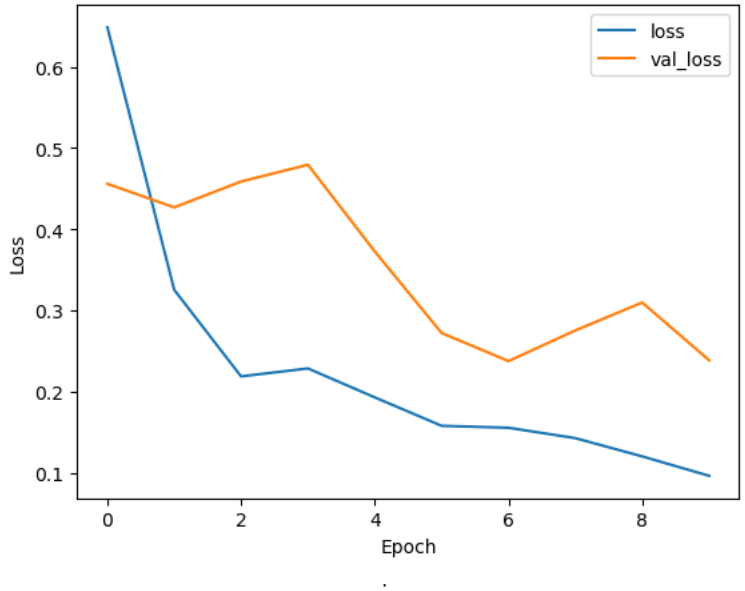
Computational Results

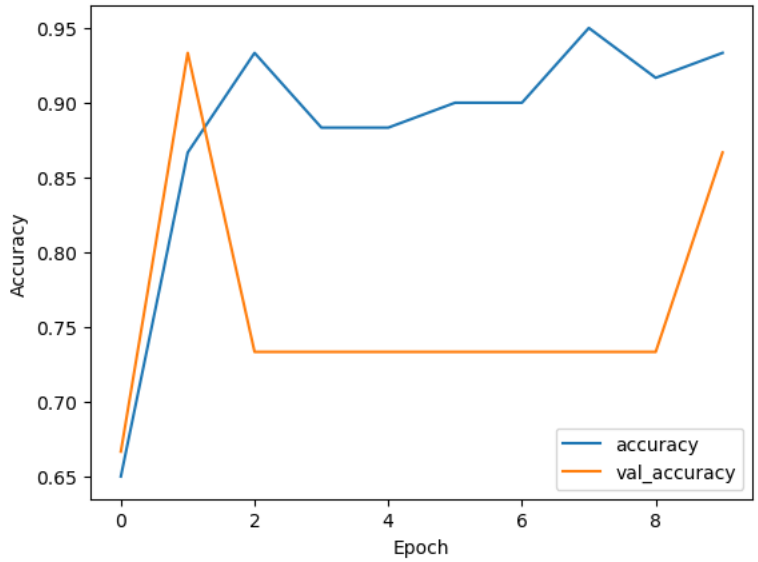
The computation results are tabulated for each Classification as below.

1. **Binary Classification Model :**

Models	Binary Classification (Amecro and BlueJay)							
	Number of Convolution Layers	**Max_Pooling Size**	**Dense**	**DropOut**	**Compilation**	**Epoch**	**Batch Size**	**Accuracy**
Model 1	**1**. 32 filters, 3*3 kernel, relu, **2**. 64 filters, 3*3 kernel, relu **3.** 128 filters, 3*3 kernel, relu	**1.** 2*2 **2.** 2*2 **3.** 2*2	**1.** 128 units, relu **2.** 1 unit, sigmoid	1 - rate : 0.5 (Before 1st Dense)	**optimiser** : rmsprop **loss** :binary_crossentropy **Metrics**: Accuracy	20	75	92.00 %
Model 2	**1.** 32 filters, 3*3 kernel, relu, **2**. 64 filters, 3*3 kernel, relu	**1.** 2*2 **2.** 2*2	**1.** 128 units, relu **2.** 1 unit, sigmoid	1 - rate : 0.4 (Before 1st Dense)	**optimiser** : rmsprop **loss** :binary_crossentropy **Metrics**: Accuracy	10	20	96.00 %
Model 3	**1**. 32 filters, 3*3 kernel, relu, **2.** 64 filters, 3*3 kernel, relu **3.** 128 filters, 3*3 kernel, relu	**1.** 2*2 **2**. 2*2 **3**. 2*2	**1.** 128 units, relu **2.** 1 unit, sigmoid		**optimiser** : adam **loss** : binary_crossentropy **Metrics**: Accuracy	10	32	96.00 %

Plots of Convergence (Model 3)

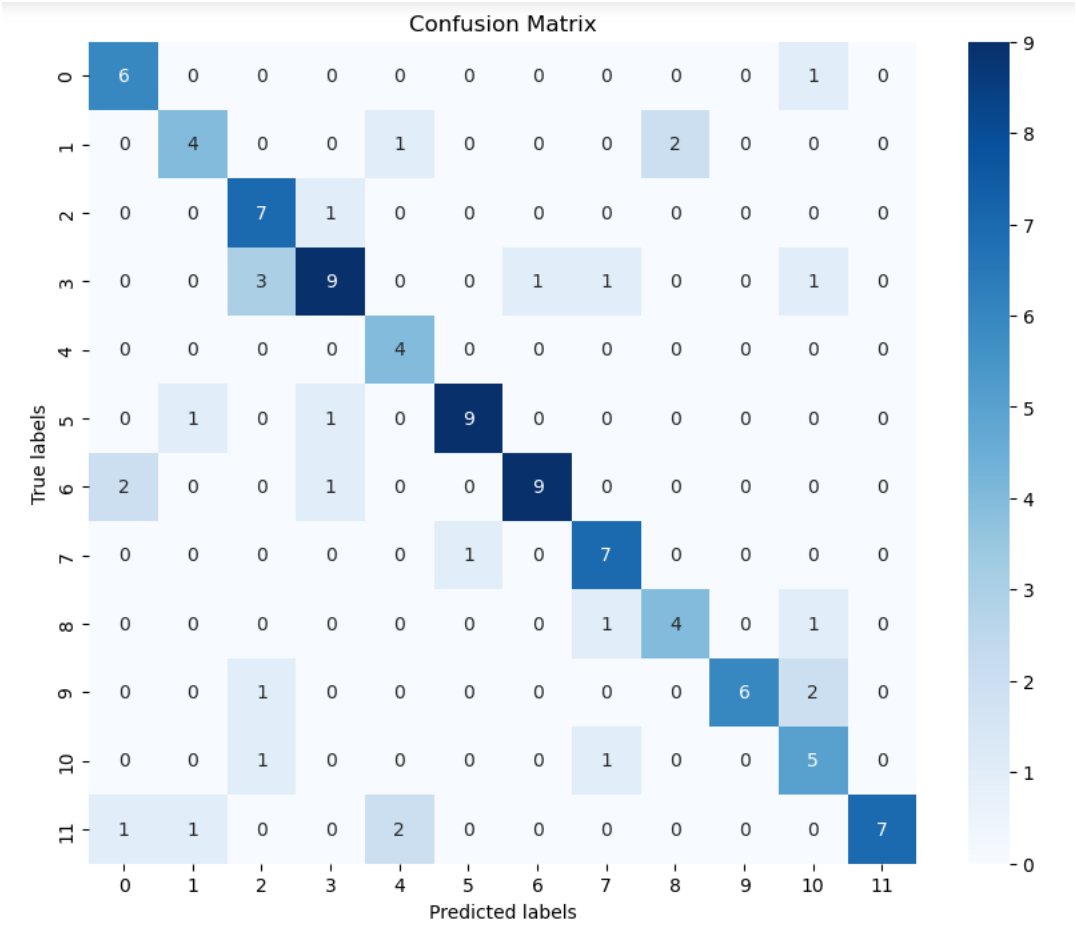




2. Multi Class Classification Model:

Models	MultiClass Classification (12 Species)							
	Number of Convolution Layers	**Max_Pooling Size**	**Dense**	**DropOut**	**Compilation**	**Epoch**	**Batch Size**	**Accuracy**
Model 1	**1.** 32 filters, 3*3 kernel, relu, **2.** 64 filters, 3*3 kernel, relu	**1.** 2*2 **2.** 2*2	**1.** 128 units, relu **2.** 12 unit, softmax		**optimiser** : adam **loss** :categorical_crossentropy **Metrics**: Accuracy	10	10	69.52 %
Model 2	**1.** 32 filters, 3*3 kernel, relu, **2.** 64 filters, 3*3 kernel, relu	**1.** 2*2 **2.** 2*2	**1.** 128 units, relu **2.** 12 unit, softmax		**optimiser** : adam **loss** : categorical_crossentropy **Metrics**: Accuracy	10	10	63.81 %
Model 3	**1.** 16 filters, 3*3 kernel, relu, **2.** 32 filters, 3*3 kernel, relu	**1.** 2*2 **2.** 2*2	**1.** 128 units, relu **2.** 12 unit, softmax	1 - rate : 0.5 (Before 1st Dense)	**optimiser** : adam **loss** :categorical_crossentropy **Metrics**: Accuracy	10	20	71.43 %
Model 4	**1.** 32 filters, 3*3 kernel, relu, **2.** 64 filters, 3*3 kernel, relu **3.** 128 filters, 3*3 kernel,relu	**1.** 2*2 **2.** 2*2 **3.** 2*2	**1.** 256 units, relu **2.** 128 units, relu **3.** 12 unit, softmax	1 - rate : 0.5 (After 1st dense)	**optimiser** : adam **loss** :categorical_crossentropy **Metrics**: Accuracy	10	15	72.38 %

Confusion Matrix of Model 4



Discussion

The field of deep learning methods for classifying bird species was investigated in this research. The objective of this research was to identify the variables that affect the precision of bird species identification through their vocalizations. By convolutional neural network (CNN) models, complex patterns present in spectrogram data were aimed to be revealed and the characteristics that differentiate various bird species were sought to be clarified.

In this study, the model-4 of multi-class classification, which achieved an accuracy of 72.38%, was employed to predict the labels for three external audio clips. Surprisingly, the model predicted all of them to be Dark-eyed Junco bird calls. Furthermore, the model's top five predictions were as follows: Dark-eyed Junco (1.0), White-crowned Sparrow (0.0), Western Meadowlark (0.0), Stellar's Jay (0.0), and Red-winged Blackbird (0.0). These findings suggest that while the model demonstrated high accuracy on the training and validation datasets, it encountered challenges when applied to unseen

external data, potentially indicating limitations in its generalization capabilities or biases in the training data. Further investigation into the reasons behind these discrepancies is warranted to refine the model's performance and enhance its reliability in real-world applications.

Test Spectrogram	Top 5 Predictions
Spectrogram 1	1. daejun 2. whcspa 3. wesmea 4. stejay 5. rewbla
Spectrogram 2	1. daejun 2. whcspa 3. wesmea 4. stejay 5. rewbla
Spectrogram 3	1. daejun 2. whcspa 3. wesmea 4. stejay 5. rewbla

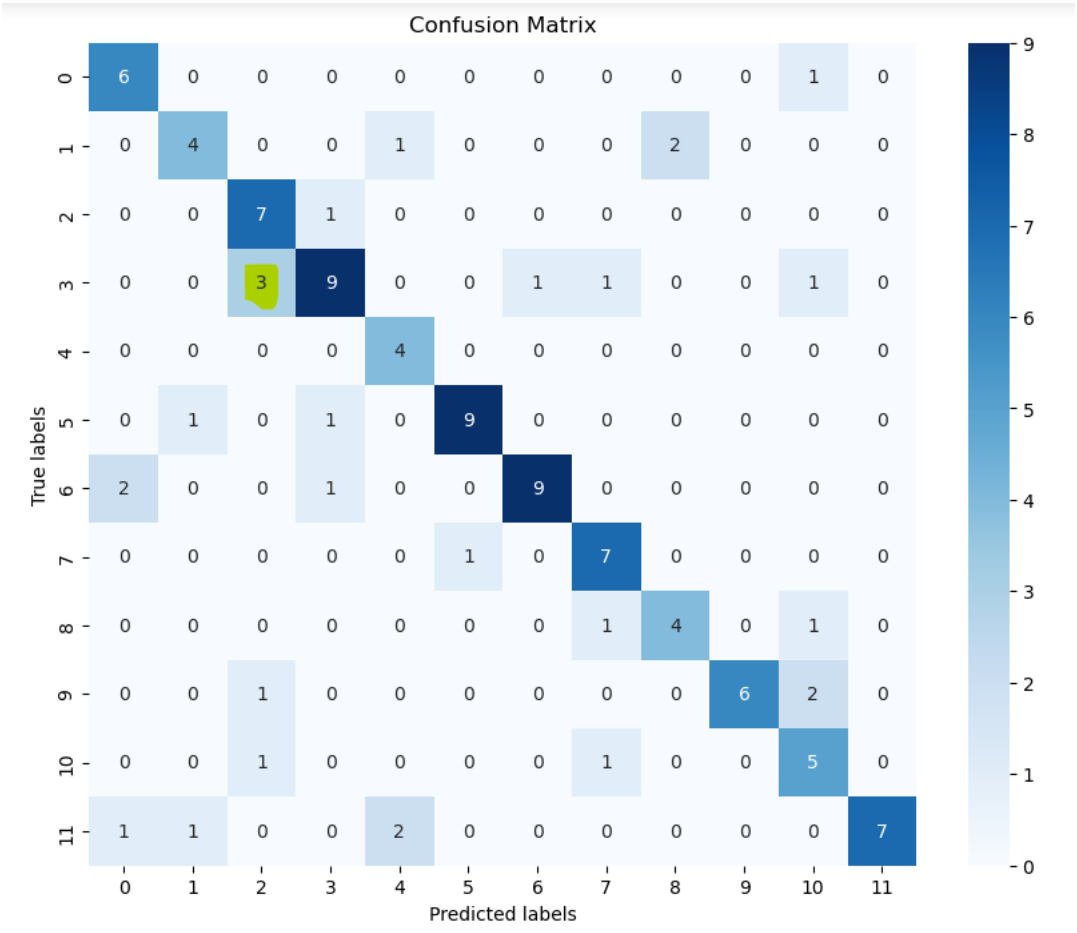
When utilizing Model 3 from multi class for prediction, the dominant prediction across all three spectrograms is Northern Flicker (norfli), with a confidence score of 1.0 for each prediction. Additionally, the top five predictions for each spectrogram remain consistent, with Northern Flicker occupying the first position and the remaining four species (White-crowned Sparrow, Western Meadowlark, Stellar's Jay, and Red-winged Blackbird) receiving confidence scores of 0.0. This indicates a strong bias towards Northern Flicker in the model's predictions, suggesting potential limitations in the model's ability to generalize across different bird species. Further investigation is necessary to understand the underlying reasons for this bias and to improve the model's performance on unseen data.

Test Spectrogram	Top 5 Predictions
Spectrogram 1	1. norfli 2. whcspa 3. wesmea 4. stejay 5. rewbla
Spectrogram 2	1. norfli 2. whcspa 3. wesmea 4. stejay 5. rewbla
Spectrogram 3	1. norfli 2. whcspa 3. wesmea 4. stejay 5. rewbla

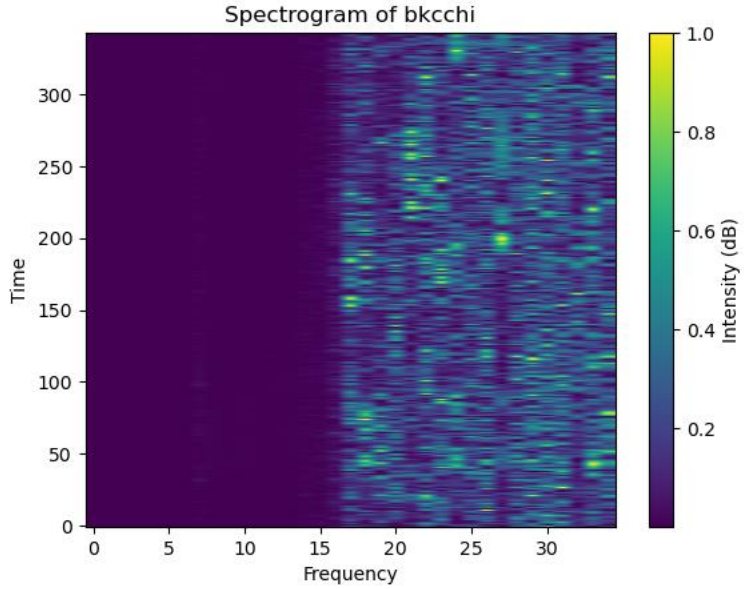
The major limitation that this study had was the above bias seen during external data prediction. Though the model's accuracy was considerable and the number of samples from each species was equally taken(35 each),the above bias was observed. To address potential biases in the model's predictions, further steps such as shuffling the data or employing cross-validation techniques can be considered. Shuffling the data might help in randomizing the order of samples and preventing the model from learning spurious correlations based on the order of the data. Additionally, cross-validation techniques (Hyper parameter tuning) can be utilized to evaluate the model's performance on multiple subsets of the data, thereby providing a more robust assessment of its generalization capabilities. By implementing these additional steps, it may be possible to uncover and mitigate any biases present in the model's predictions, ultimately improving its reliability and accuracy.

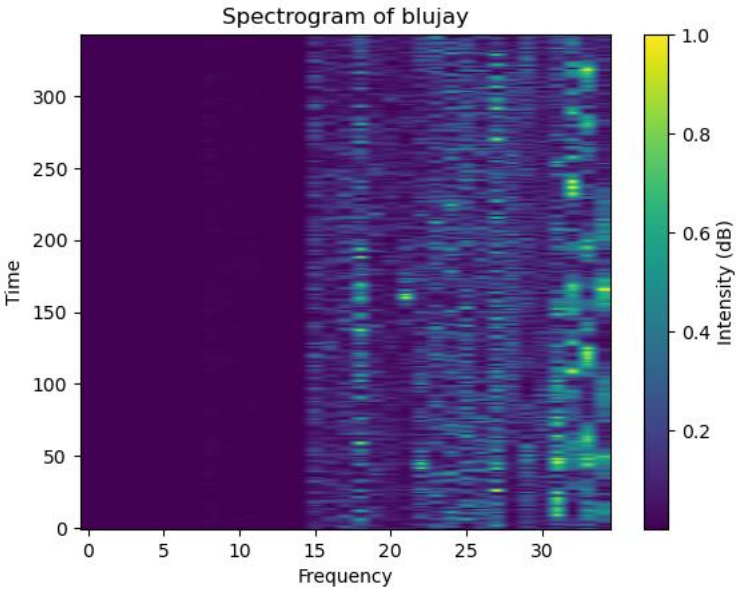
The models took about 15 to 20 mins on an average to be trained. For every model, initially smaller samples were considered and also smaller values of epochs were considered. Then the final accuracy was calculated using the full data set and bigger epochs.

Upon evaluating the confusion matrix of the best performing multi-class model, a notable trend was observed where the model frequently confuses between Black-capped Chickadee (bkcchi) and Blue Jay (blujay) species. Specifically, there were approximately three instances where Blue Jays were incorrectly predicted as Black-capped Chickadees.



Upon further examination by plotting and analyzing the spectrograms of these two species, it became apparent that their spectrograms exhibited similarities in frequency range distribution and appearance. This similarity in spectrogram characteristics likely contributed to the model's difficulty in distinguishing between the two species, highlighting the importance of considering the unique acoustic features of each bird species in improving classification accuracy.





In this application of bird species classification based on vocalizations, both Random Forest and Support Vector Machine (SVM) models offer viable alternatives to neural networks. Random Forest is an ensemble learning method capable of handling high-dimensional data and is known for its robustness and ability to handle noisy or correlated features. Similarly, SVM is a powerful supervised learning algorithm effective in high-dimensional spaces, capable of finding optimal hyperplanes to separate classes

in the feature space. Both models have been successfully applied in various classification tasks and offer interpretable results, making them attractive choices for bird species classification.

However, neural networks, particularly convolutional neural networks (CNNs), offer distinct advantages in this application. Unlike Random Forest and SVM, CNNs are specifically designed to handle spatial data such as spectrograms, allowing them to automatically learn hierarchical representations of features directly from the raw data. This feature learning capability reduces the need for manual feature engineering and enables CNNs to capture complex patterns and spatial relationships inherent in bird vocalizations. Additionally, CNNs exhibit strong generalization capabilities and can scale effectively to handle large datasets and complex classification tasks, making them well-suited for real-world applications with diverse input data.

Conclusion

This study explored the application of deep learning techniques for the classification of bird species based on their vocalizations. Convolutional neural networks (CNNs) trained on spectrogram data were leveraged with the aim of accurately identifying bird species from audio recordings. Promising results were achieved through meticulous preprocessing and model training, with the best-performing model demonstrating a commendable accuracy of 72.38%. However, challenges were revealed in the analysis, including biases towards certain species and confusion between similar spectrogram patterns. These findings highlighted the complexity of the task and the need for further investigation into model refinement and data augmentation techniques to improve classification accuracy. Additionally, the study identified potential areas for future research, such as exploring alternative model architectures and cross validating and tuning different hyper parameters. The growing body of research in bioacoustics benefits from this study, highlighting the potential of deep learning methods to advance in comprehension of avian vocalizations and biodiversity monitoring.

Bibliography/References:

1. **https://xeno-canto.org/**
2. https://www.researchgate.net/figure/Deep-learning-diagram_fig5_323784695