REPORT ON

CLASSIFICATION OF BIRD SPECICES BASED ON THEIR SOUND USING NEURAL NETWORKS

ABSTARCT

In this report, neural networks are used to classify bird species based on their vocalizations. It uses a dataset of spectrograms from audio recordings of 12 bird species, which are sourced from Xeno-Canto.[1] Additionally, external testing was performed on three MP3 bird call recordings. The study focuses on two tasks: a binary classification model distinguishing between the House sparrow and Song sparrow, and a multi-class classification to identify any of the 12 species. Predictions on the three external test clips are made to assess the effectiveness of the models. For hidden layers, various activation functions such as ReLU, SoftMax, and ELU were used. Two convolutional neural network models were developed and evaluated with different architectures and parameters, including dropout and regularization techniques. This report concludes with discussion of model limitations, challenges in distinguishing similar bird calls, and potential alternative approaches for this application.

INTRODUCTION

In this study, neural networks were used to classify bird species based on their vocalizations. The dataset consists of pre-processed spectrograms generated from audio recordings collected in the Seattle area, sourced from the Xeno-Canto[1] Birdcall competition. These spectrograms, which visually represent the frequency and intensity of bird calls over time, were stored in HDF5 format and used as input to train the models.

The main goal of this project was to develop a neural network that could accurately predict bird species based on their unique sound patterns. To do this, we performed both binary and multi-class classification tasks and compared different model architectures and hyperparameters to find the most effective approach. We also discuss the limitations faced during the process.

In addition to the main dataset, we tested the model on three external MP3 bird call recordings. These audio files were converted into spectrograms using the same preprocessing method and analyzed using the trained neural networks.

Throughout the project, we encountered challenges such as data imbalance, variations in spectrogram shapes, and difficulty distinguishing between species with similar vocalizations. This report explores these challenges and how they affected model performance. These findings highlight the potential of neural networks in bird species identification and their possible implications for conservation efforts.

THEORITICAL BACKGROUND

NEURAL NETWORKS

Neural networks are a class of machine learning algorithms inspired by the structure and functioning of the human brain. They consist of interconnected nodes, called neurons, which are organized into multiple layers. These networks process information by passing data through these layers, allowing them to learn and recognize patterns. In the context of bird species identification, neural networks can be trained to detect and learn the unique patterns and features of each species based on their distinctive sounds, enabling accurate classification and prediction.

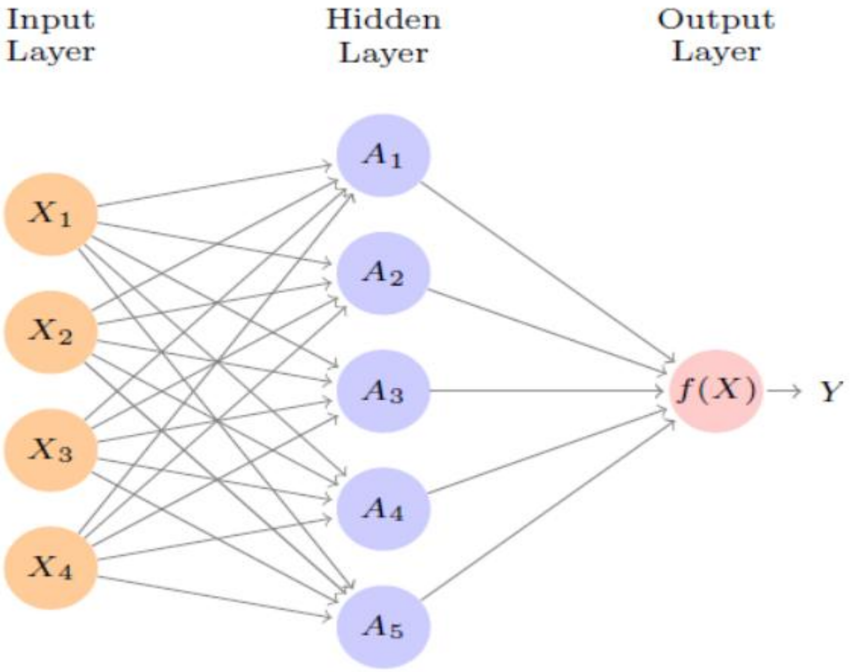


Figure -1 Architecture of Neural Network with 3 layers

From figure 1 Architecture of Neural Network has three layers. The first layer is the input layer, where we have four input nodes labeled X_1, X_2, X_3, and X_4. Each of these represents a feature provided to the network. These inputs are sent to the hidden layer, which contains five neurons labeled A_1, A_2, A_3, A_4, and A_5. Each neuron in the hidden layer takes signals from all four inputs, does some calculations, and passes the results forward. Finally, the information flows into the output layer, where a final calculation, labeled f(X), produces the output Y, which could be a prediction or classification result.

Activation Functions: It is a key part of each neuron in a neural network. After the neuron calculates the weighted sum of inputs, the activation function transforms this number into an output that is passed to the next layer. Without activation functions, the neural network would only be able to learn to be in linear patterns, no matter how many layers it has. By using activation functions like ReLU, Sigmoid, or Tanh, the network can learn more complex and non-linear patterns in the data. For example, in this network, each hidden layer neuron (A_1, A_2, A_3, A_4, A_5) applies an activation function to its input to decide how much signal to pass forward.

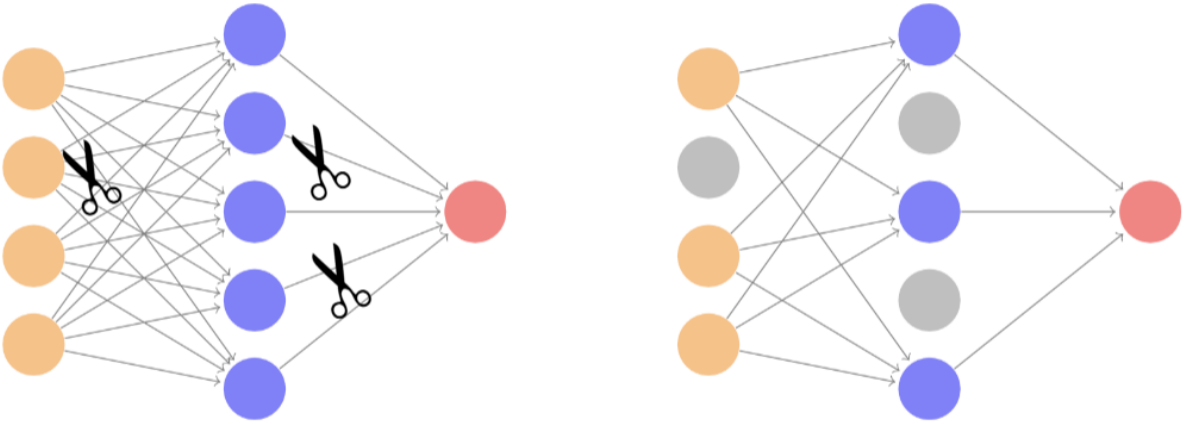


Figure -2 Dropout Layers for Regularization

From figure-2 dropout layer is a regularization technique that works by randomly dropping or deactivating some neurons inside the network during training to prevent overfitting. In the left side of the image, the scissors show how certain connections and neurons are randomly removed and ignored, which means they don't contribute to that training step. On the right side, the grayed-out nodes represent the dropped neurons, while the remaining active neurons continue passing information forward. By dropping different neurons each time, the network learns to not depend too much on any single neuron, which improves its ability to generalize new data.

Convolutional neural networks (CNN)

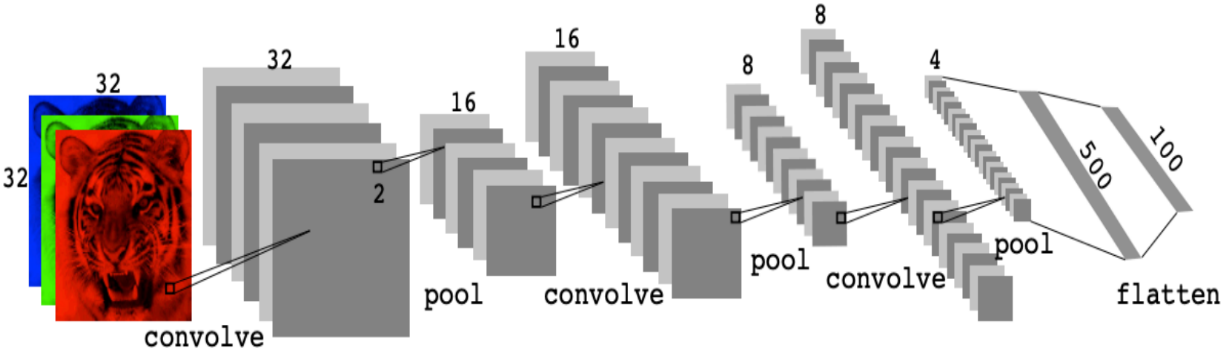


Figure-3 Architecture of Convolutional Neural Network for Image Classification

Convolutional Neural Networks (CNNs) are deep learning models designed for structured, grid-like data such as images or audio spectrograms. They work by automatically learning features through a series of layers. As shown in figure 3, the CNN processes an image step by step. On the far left, the input image of a tiger is 32x32 pixels. The process begins with a convolution layer, where filters into small patterns scan across the image to detect simple features like edges or textures, producing a set of feature maps.

Next, a pooling layer reduces the size of these maps while preserving the most important information, making the model more efficient and less sensitive to small changes in the input. This sequence repeats where another convolution layer detects more complex patterns, followed by pooling to reduce size again. As the image moves deeper into the network, it becomes smaller but captures increasingly abstract features. Finally, the resulting feature maps are flattened into a one-dimensional vector and passed into fully connected layers to make the final prediction. This architecture enables CNNs to automatically extract and combine simple to complex features, making them highly effective for image or spectrogram classification tasks.

METHODOLOGY

This research aimed to classify bird sounds into different species using spectrogram data stored in an HDF5 file. The dataset[1] contained audio features from 12 bird species, each represented as spectrogram arrays of varying sample sizes and shapes. Where each species originally shaped as 128 frequency bins by 517-time steps. To prepare the data for neural network input, the spectrograms were transposed and resized to 256 frequencies by 343-time dimensions using anti-aliasing techniques. This resizing step ensured that all inputs shared a consistent shape compatible with convolutional neural network models.

For binary classification, we focused on distinguishing between the House Sparrow and Song Sparrow classes. The processed spectrograms were normalized to a [0,1] scale and expanded with a channel dimension to match the expected input shape for CNNs. Labels were assigned as 0 for House Sparrow and 1 for Song Sparrow. The dataset was split into training of 80% and testing of 20% using stratified sampling to maintain class balance.

We trained three binary classification models with different architectures to explore model performance. The first model was a simple CNN with two convolutional layers, ReLU activation, max pooling, a dense layer, and dropout regularization. The second model increased the dropout rate to 0.7 to reduce overfitting, while the third model included both dropout and L2 regularization along with early stopping to improve generalization. All models were trained using the Adam optimizer, binary cross-entropy loss, and evaluated based on accuracy, confusion matrix, and classification report.

For multi-class classification, we included all 12 bird species. Twenty spectrogram samples were selected per species to ensure balanced representation were resized to a uniform shape 256 × 343 and normalized for input into convolutional neural networks. The species labels were encoded using one-hot encoding to prepare for multi-class classification. We split the dataset into 80% training and 20% testing subsets using stratified sampling.

We developed three multi-class CNN models to compare performance. The first model included two convolutional layers, ReLU activations, and a dropout layer. The second model added L2 regularization to the convolutional and dense layers to reduce overfitting. The third model introduced an additional convolutional layer to learn deeper features. Each model was trained for

20 epochs using the Adam optimizer and categorical cross-entropy loss, with performance evaluated on validation and test sets. All models performed test accuracy, confirmed through confusion matrices and classification reports, precision, recall, and F1-scores across all classes.

For external testing, we evaluated the trained multi-class CNN model on unseen bird audio recordings by extracting the three loudest 2-second segments from each recording using energy-based segmentation and RMS. The spectrograms were saved as grayscale images, normalized, and fed into the trained model for prediction. For each recording, the model produced probability scores for each species across the three spectrograms. These probabilities were averaged to obtain a final prediction. We reported the top three predicted species for each recording based on the highest averaged probabilities, enabling robust evaluation on external data.

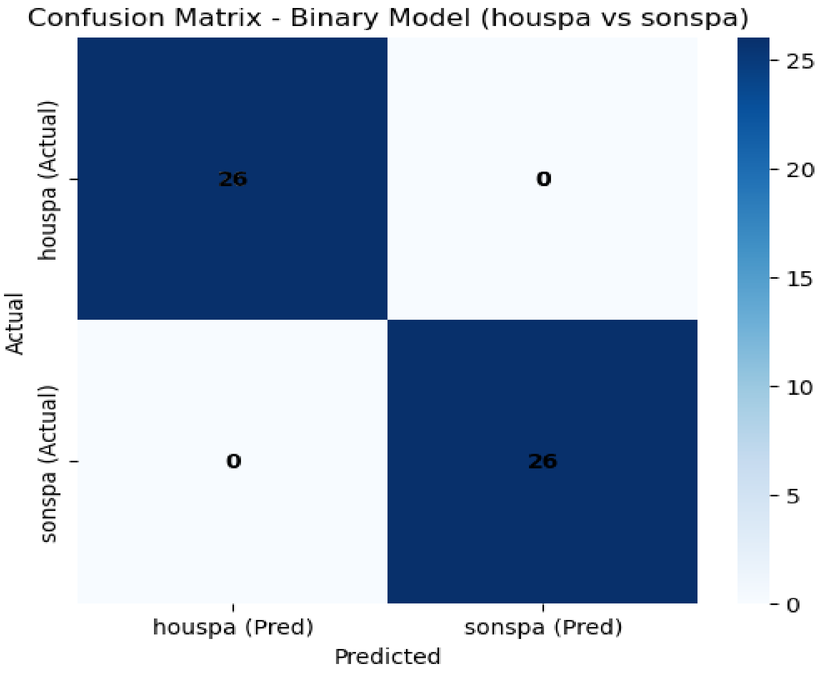
COMPUTATIONAL RESULTS

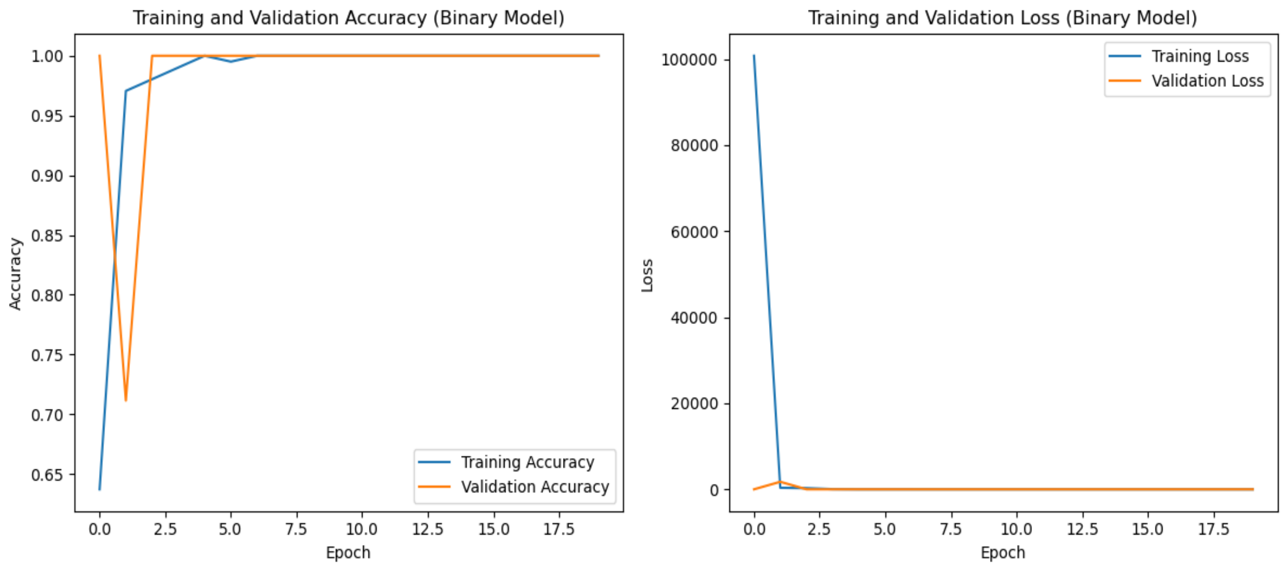
BINARY CLASSIFICATION

Model evaluation:

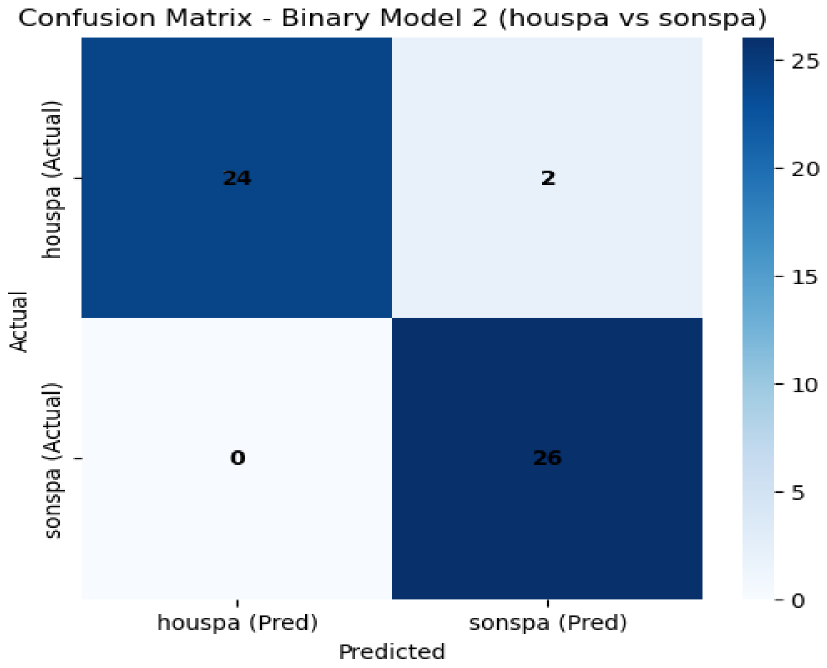
Model	Conv Layers	Dense Layers	Dropout	Regularization L2	Epochs	Validation Accuracy (%)	Test Accuracy (%)
Binary Model 1	2	1	0.5	None	20	100.00	100.00
Binary Model 2	2	1	0.7	None	20	96.15	96.15
Binary Model 3 (L2)	2	1	0.5	0.001	7*	100.00	100.00
Binary Model 4 (ELU)	2	1	0.5	0.001	9*	73.08	73.08

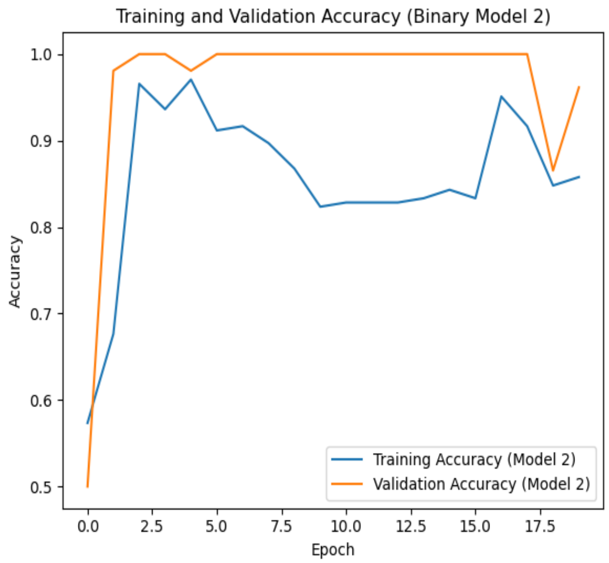
Model-1: Baseline CNN with Dropout 0.5: The baseline CNN model with 2 convolutional layers, 1 dense layer, and a dropout rate of 0.5 achieved 100% validation accuracy and 100% test accuracy, indicating very strong performance. The training and validation curves remained stable after the initial epoch, suggesting good generalization despite no explicit regularization other than dropout. However, the zero-validation loss from epoch 5 onward indicates potential overfitting to the small validation set.

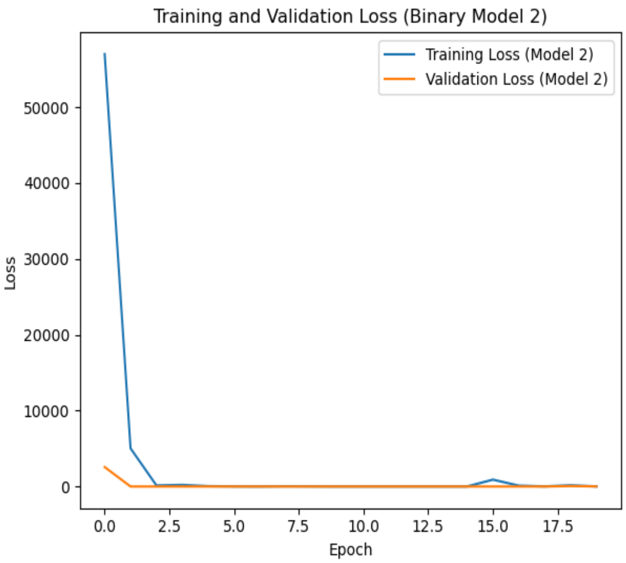




Model 2: Increased Dropout 0.7: Increasing the dropout rate to 0.7 slightly reduced performance, with a final validation and test accuracy of 96.15%. This suggests that while dropout helped regularize the model, a high dropout rate suggests that too much removal of information during training, limiting performance.

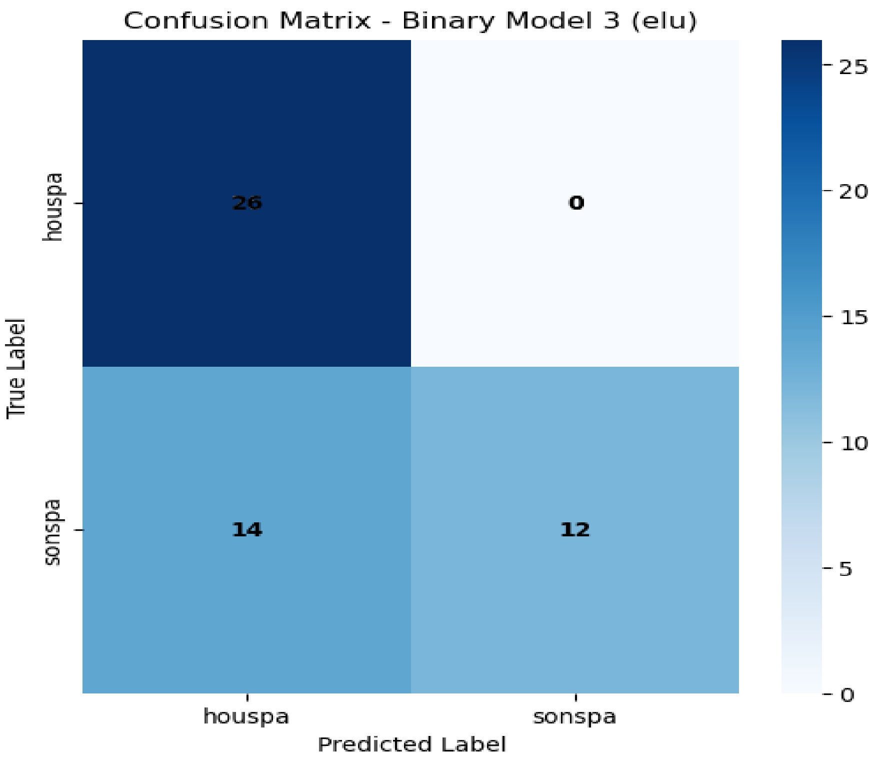


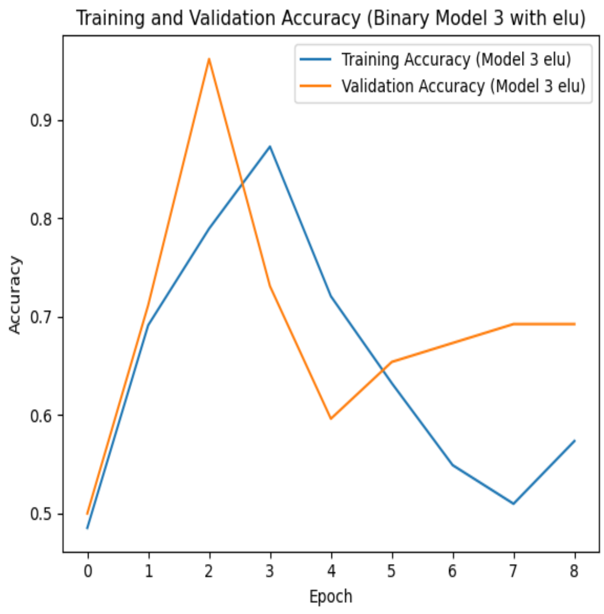


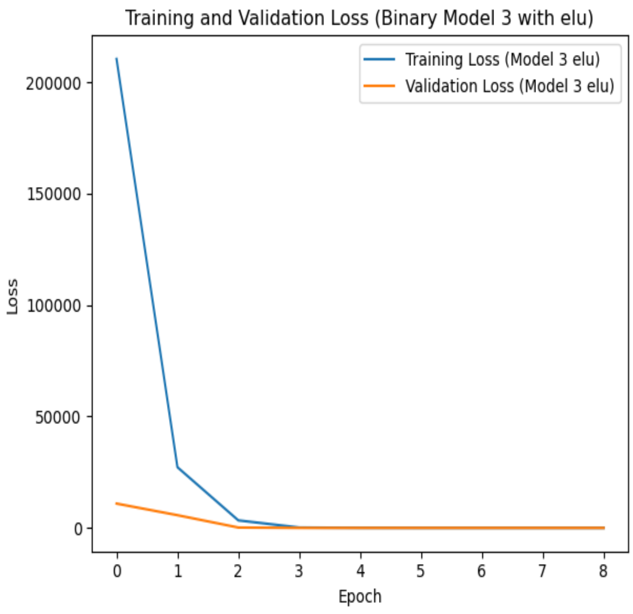


Model-3: Dropout + L2 Regularization + Early Stopping: Incorporating L2 regularization (λ = 0.001) a long dropout (0.5) and early stopping resulted in 100% validation accuracy with early stopping activated at epoch 7. The model achieved 100% test accuracy, showing strong generalization while avoiding overfitting. Validation loss remained low and stable, suggesting the combined regularization effectively controlled complexity.

Model-4:ELU Activation + L2 + Dropout + Early Stopping: Switching to the ELU activation function under the same architecture reduced validation accuracy to 73.08%, with the model showing higher validation loss and earlier plateauing. This suggests the ELU activation was less effective for this dataset compared to ReLU under similar regularization settings.







Classification Performance Comparison-Binary Classification

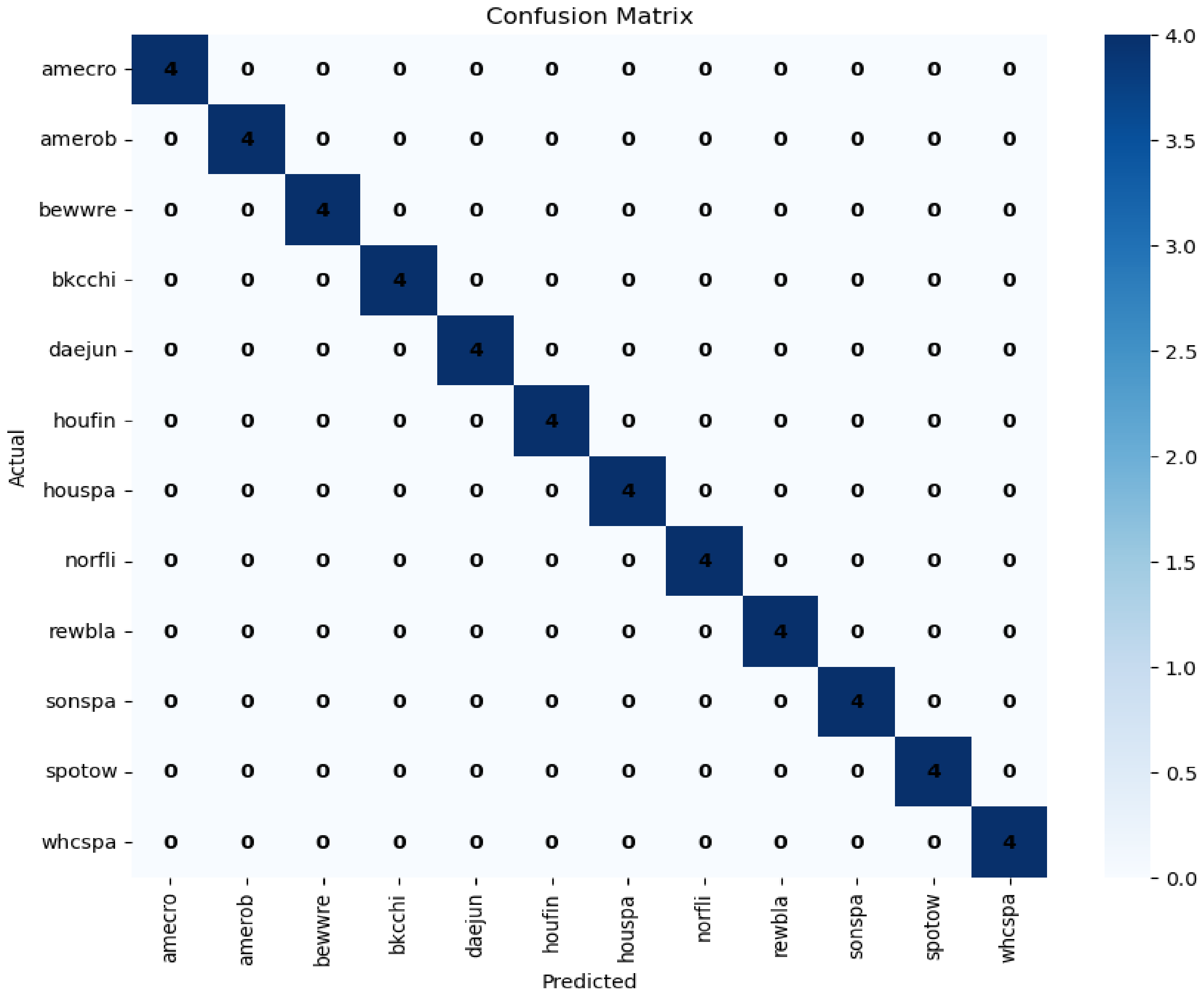
Metric	Model 1	Model 2	Model 3	Model 4
Precision	1.00	0.96	1.00	0.82
Recall	1.00	0.96	1.00	0.73
F1-score	1.00	0.96	1.00	0.71

From the Classification Performance of binary classification, the precision, recall, and F1-score for the four binary classification models. Model 1 and Model 3 achieved perfect precision, recall, and F1-score 1.00, indicating excellent performance with no misclassifications on the test set. Model 2 performed similarly well, with slightly lower precision and recall at 0.96. In contrast, Model 4 showed noticeably lower performance, achieving a precision of 0.82, recall of 0.73, and F1-score of 0.71, suggesting this model had difficulty correctly identifying both classes compared to the other models.

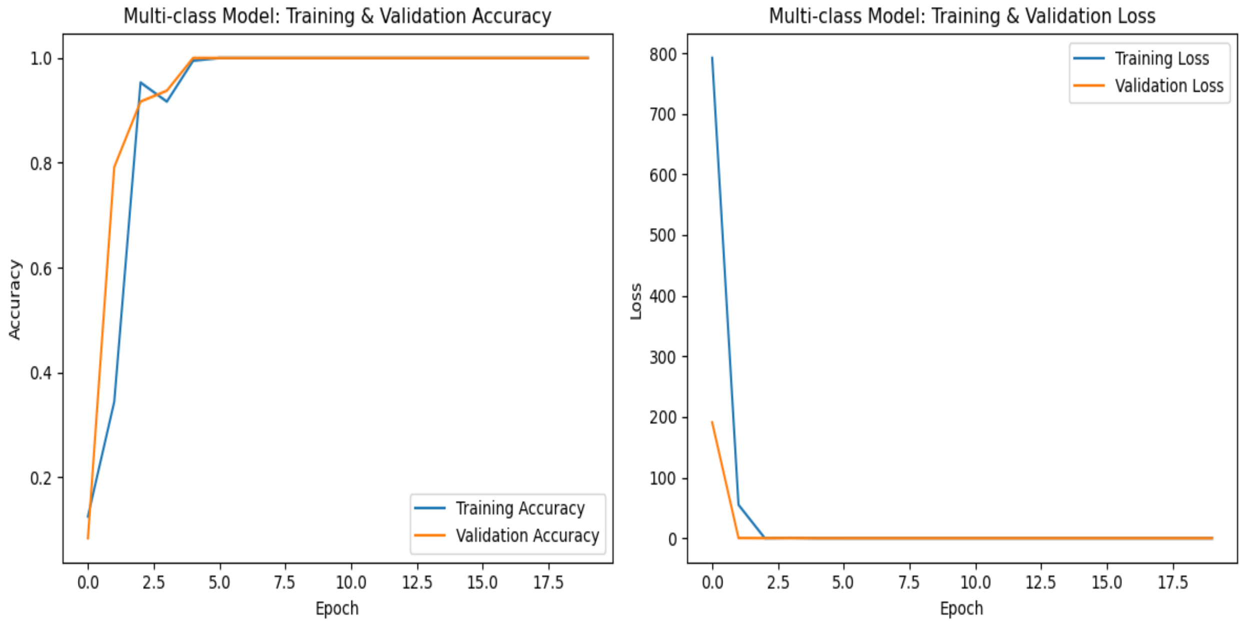
MULTI-CLASS CLASSIFICATION:

Model	Conv Layers	Dropout	Regularization L2	Epochs	Validation Accuracy (%)	Test Accuracy (%)
Model 1: Baseline CNN	2	0.5	None	20	100.00	100.00
Model 2: CNN + Dropout + L2	2	0.5	0.001	20	100.00	100.00
Model 3: CNN with 3 Conv Layers	3	0.5	None	20	100.00	100.00

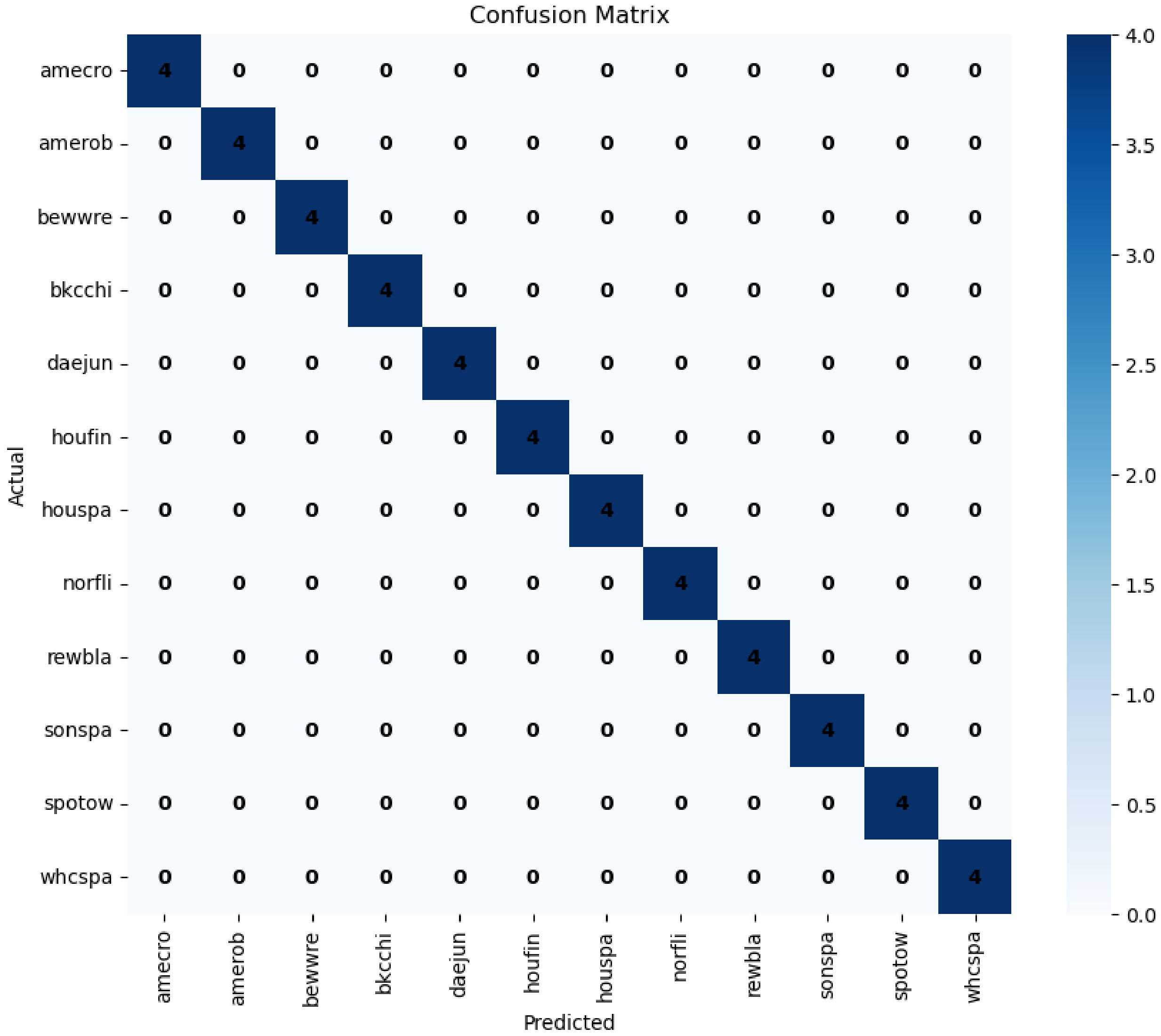
Model-1 CNN models used 2 convolutional layers, a dropout rate of 0.5, and n. The model achieved a validation accuracy of 100% and a test accuracy of 100% after 20 epochs, indicating excellent performance.

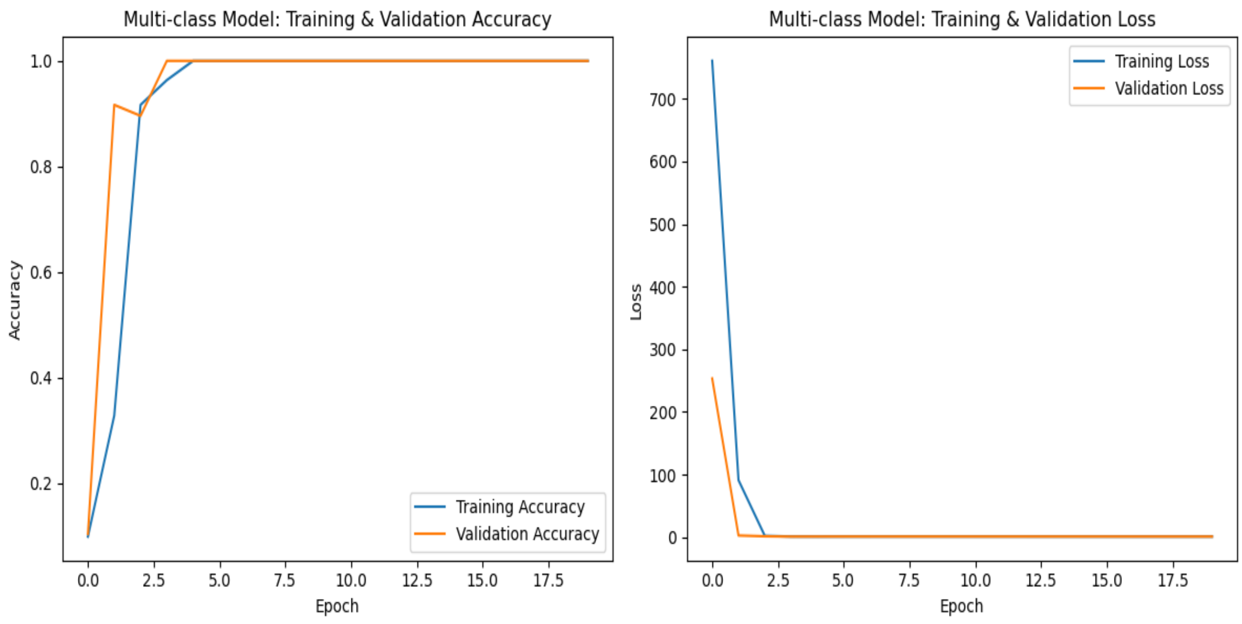


	amecro	amerob	bewwre	bkcchi	daejun	houfin	houspa	norfli	rewbla	sonspa	spotow	whcspa
amecro	4	0	0	0	0	0	0	0	0	0	0	0
amerob	0	4	0	0	0	0	0	0	0	0	0	0
bewwre	0	0	4	0	0	0	0	0	0	0	0	0
bkcchi	0	0	0	4	0	0	0	0	0	0	0	0
daejun	0	0	0	0	4	0	0	0	0	0	0	0
houfin	0	0	0	0	0	4	0	0	0	0	0	0
houspa	0	0	0	0	0	0	4	0	0	0	0	0
norfli	0	0	0	0	0	0	0	4	0	0	0	0
rewbla	0	0	0	0	0	0	0	0	4	0	0	0
sonspa	0	0	0	0	0	0	0	0	0	4	0	0
spotow	0	0	0	0	0	0	0	0	0	0	4	0
whcspa	0	0	0	0	0	0	0	0	0	0	0	4

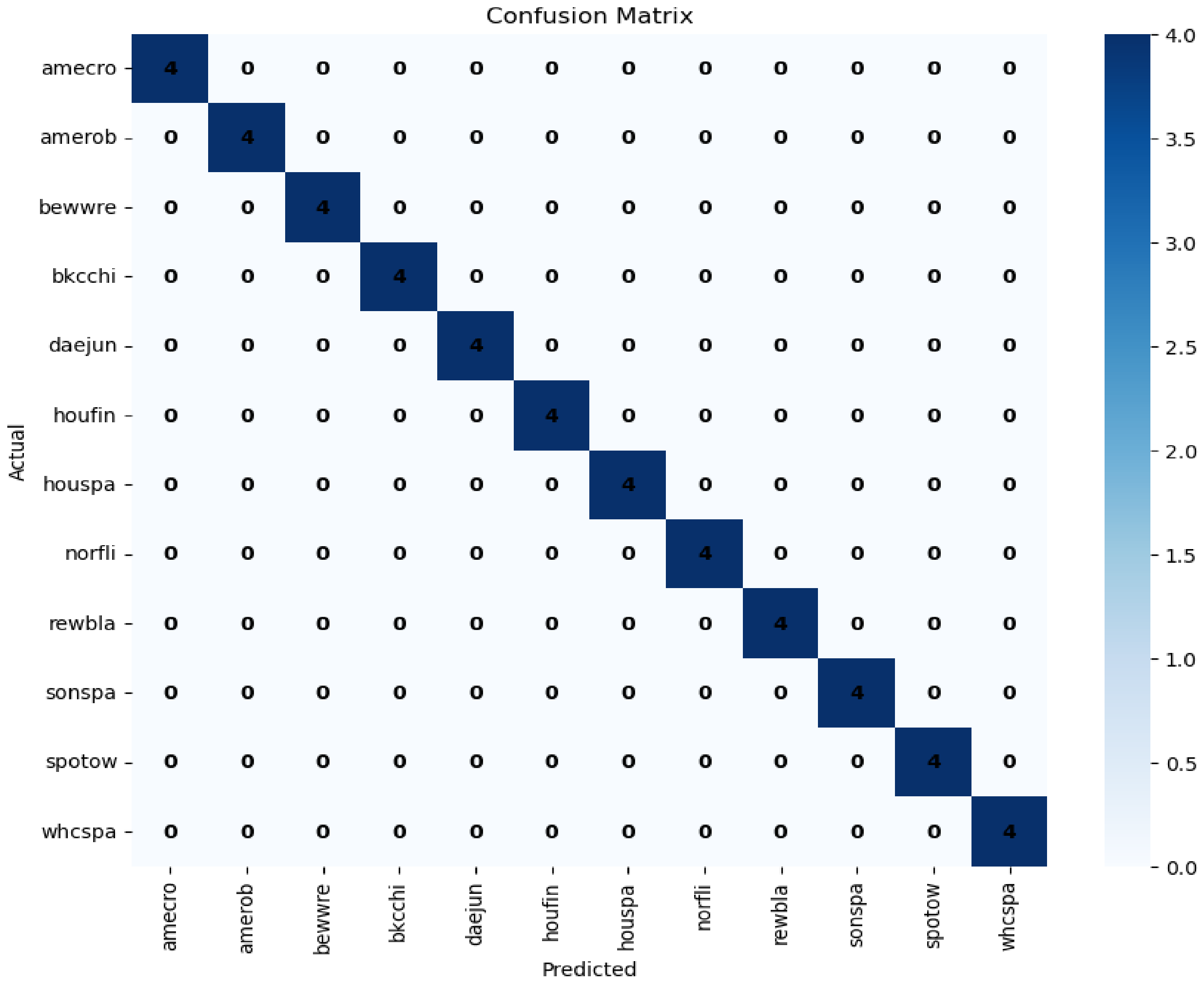


Model 2: CNN with Dropout and L2 Regularization: This has architecture of 2 convolutional layers and a dropout rate of 0.5 but added L2 regularization 0.001 to control overfitting. Like the baseline model, it also achieved 100% validation and test accuracy after 20 epochs. The validation and test losses were slightly higher ~0.93 compared to Model 1, indicating the regularization effect increased the loss term without hurting classification performance. This suggests the model achieved strong generalization with added robustness from L2 regularization.

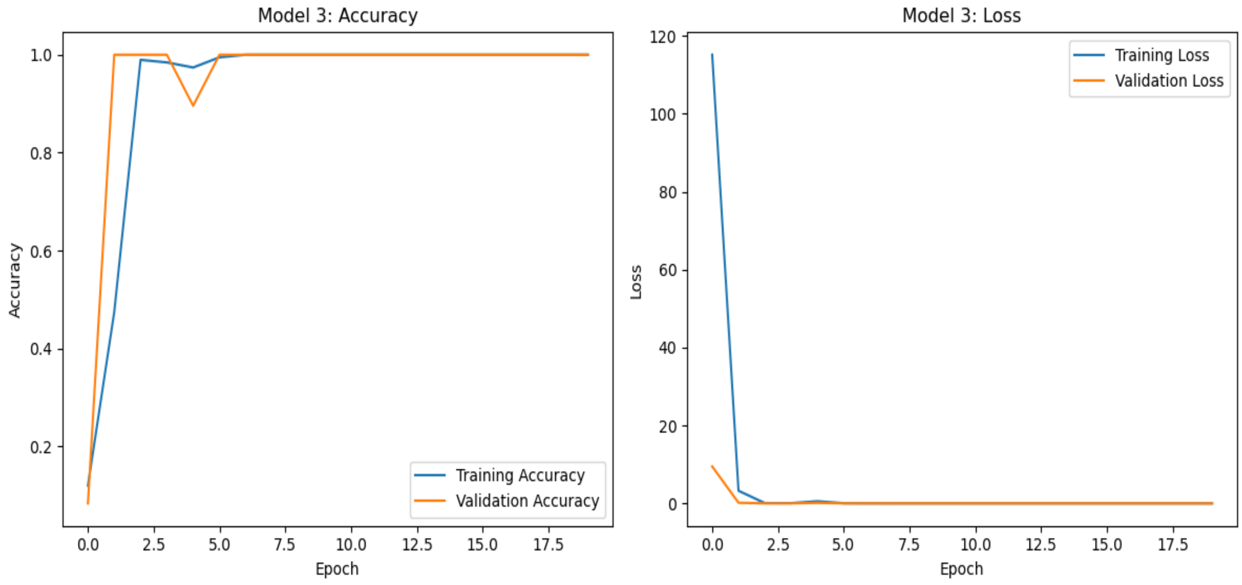




Model 3: CNN with 3 Convolutional Layers: In this model, the architecture was made deeper by using 3 convolutional layers, while maintaining the dropout rate of 0.5 and no other regularization. Similar to the other models, it reached 100% validation and test accuracy after 20 epochs, with nearly zero validation and test losses. The deeper architecture provided no additional improvement in accuracy but maintained perfect performance, showing that increased depth did not negatively impact the model.



	amecro	amerob	bewwre	bkcchi	daejun	houfin	houspa	norfli	rewbla	sonspa	spotow	whcspa
amecro	4	0	0	0	0	0	0	0	0	0	0	0
amerob	0	4	0	0	0	0	0	0	0	0	0	0
bewwre	0	0	4	0	0	0	0	0	0	0	0	0
bkcchi	0	0	0	4	0	0	0	0	0	0	0	0
daejun	0	0	0	0	4	0	0	0	0	0	0	0
houfin	0	0	0	0	0	4	0	0	0	0	0	0
houspa	0	0	0	0	0	0	4	0	0	0	0	0
norfli	0	0	0	0	0	0	0	4	0	0	0	0
rewbla	0	0	0	0	0	0	0	0	4	0	0	0
sonspa	0	0	0	0	0	0	0	0	0	4	0	0
spotow	0	0	0	0	0	0	0	0	0	0	4	0
whcspa	0	0	0	0	0	0	0	0	0	0	0	4



EXTERNAL TESTING:

From the table it shows the predicted probabilities of each class of species for each segment of each external audio file. predicted probabilities for each bird species across three external test audio files, evaluated on their top 3 loudest segments

Species	Test-1			Test-2			Test-3		
	S1	S2	S3	S1	S2	S3	S1	S2	S3
American crow	0.0840	0.0842	0.0842	0.0847	0.0844	0.0841	0.0850	0.0847	0.0853
American robin	0.0796	0.0794	0.0789	0.0789	0.0787	0.0790	0.0792	0.0787	0.0790
Bewick's wren	0.0792	0.0792	0.0793	0.0789	0.0790	0.0791	0.0788	0.0790	0.0788
Black-capped chickadee	0.0829	0.0828	0.0829	0.0829	0.0831	0.0830	0.0835	0.0837	0.0832
Dark-eyed junco	0.0833	0.0833	0.0834	0.0838	0.0835	0.0836	0.0835	0.0835	0.0835
House finch	0.0847	0.0848	0.0850	0.0850	0.0853	0.0852	0.0846	0.0851	0.0849
House sparrow	0.0839	0.0840	0.0836	0.0838	0.0839	0.0841	0.0832	0.0833	0.0833
Northern flicker	0.0854	0.0851	0.0852	0.0851	0.0858	0.0855	0.0851	0.0854	0.0856
Red-winged blackbird	0.0856	0.0857	0.0856	0.0853	0.0853	0.0850	0.0861	0.0857	0.0856
Song sparrow	0.0816	0.0815	0.0816	0.0820	0.0812	0.0814	0.0813	0.0816	0.0812
Spotted towhee	0.0836	0.0838	0.0839	0.0838	0.0836	0.0837	0.0834	0.0831	0.0833
White-crowned sparrow	0.0862	0.0863	0.0864	0.0859	0.0861	0.0862	0.0862	0.0861	0.0863

Top 3 Predicted Species per Audio File Ranked by Average Probability

Rank	Test1	Test2	Test3
1	White-crowned sparrow (0.0863)	White-crowned sparrow (0.0861)	White-crowned sparrow (0.0862)
2	Red-winged blackbird (0.0856)	Northern flicker (0.0855)	Red-winged blackbird (0.0858)
3	Northern flicker (0.0853)	Red-winged blackbird (0.0852)	Northern flicker (0.0854)

From the top 3 predicted bird species for each external test audio file, ranked by average probability across the three loudest segments. Across all three audio files, the White-crowned sparrow consistently had the highest average probability 0.086, followed by Red-winged blackbird and Northern flicker.

DISCUSSION

In this study I explored CNN architecture for both binary and multi-class classification of bird species. One of the main challenges I encountered was that the models achieved perfect accuracy on both the validation and test sets. While this sounds impressive, it points to a limitation: the model has overfitted to the small dataset. Even though I tried adding dropout and L2 regularization to reduce overfitting, the models still performed perfectly on the known data, suggesting they had memorized the patterns rather than truly learned to generalize. Another challenge was the limited variety in the dataset, which made models easier to learn but made noisy environments. On the other side, the models were quickly trained, the binary classification models trained in few minutes, and the multi-class models took more than 7 minutes for 20 epochs. Increasing the number of convolutional layers and adding regularization didn't increase the training time too much.

When it comes to predicting challenging species, the situation was a bit unusual. In both binary and multi-class tasks, the models performed perfectly on the test data, meaning there were no misclassifications in the confusion matrix. However, things changed when I tested the model on external audio files. In those tests, the predicted probabilities for all species were very similar and low, usually around 0.08 to 0.09 for each species. This showed that the model struggled to confidently identify the species from the new and unseen audio. I suspect this happened because the external audio had background noise or different sound quality compared to the training data. Additionally, some bird species may have similar sounding calls, and their spectrograms might look alike. For example, species like the Red-winged Blackbird and Northern Flicker had overlapping frequency patterns, making them harder for the model to tell apart. Even visually, their spectrograms shared similarities in certain frequency ranges.

Looking at other possible approaches, I could have also tried models like random forests, support vector machines (SVMs), or k-nearest neighbors. These models work well if we first extract features like other handcrafted audio features from the sound files. However, I believe that using a neural network, especially a convolutional neural network (CNN), makes sense for this problem. CNNs are designed to process images, and in this case, spectrograms are image representations of audio. CNN have the ability to learn patterns automatically in the frequency. This makes neural networks a powerful and flexible choice for classifying bird calls based on spectrogram data.

CONCLUSION

In conclusion, this study explored both binary and multi-class classification of bird species using deep learning approaches, specifically Convolutional Neural Networks (CNNs). Through experimenting with different model architectures, including deeper CNN layers and dropout regularization, the models achieved 100% accuracy on both validation and test sets for both classification tasks. While these results indicate strong performance on internal data, external testing with new audio samples showed lower prediction confidence, suggesting the models may not generalize well to unseen data.

Overall, the findings highlight the potential of neural network models in bird species classification. These results suggest that similar approaches could support ecological research and conservation efforts by enabling automated bird species monitoring. With further development, such as using larger and more diverse datasets and better techniques for handling noisy audio, we can develop more accurate and reliable methods for identifying bird species. This would be a valuable contribution to research and conservation efforts.

REFERENCES

1. Rao, R. (n.d.). Xeno-canto bird recordings extended A-M [Dataset]. Kaggle. https://www.kaggle.com/datasets/rohanrao/xeno-canto-bird-recordings-extended-a-m
2. Hastie, T., Tibshirani, R., & James, G. (2023). An introduction to statistical learning with applications in Python. https://hastie.su.domains/ISLP/ISLP_website.pdf.download.html
3. Bastian, M. (2019, October 11). Dropout layer explained. Database Camp. https://databasecamp.de/en/ml/dropout-layer-en