**BIRD CALL CLASSIFICATION - USING DEEP LEARNING**

**By HRISHABH KULKARNI**

**Github Link:** https://github.com/Hrish-ProCoder/SML2\_Hrishabh\_Kulkarni

### **INTRODUCTION**

This project explores the application of Deep Learning to classify bird species based on short audio clips using spectrograms. The primary objective is to train models capable of accurately identifying bird calls among 12 species.

* The dataset includes .mp3 files, converted into spectrograms.
* External test data includes 3 .mp3 clips for model testing.

**Key Questions**:

* Can CNNs distinguish Seattle bird species based on their vocal patterns?
* Which model architectures and hyperparameters lead to better classification performance?

**Tools & Libraries Used**:

* Python, TensorFlow, Keras
* NumPy, Scikit-learn, Matplotlib
* Librosa for preprocessing

**Methods Applied**:

* Spectrogram generation and normalization
* CNN architecture with regularization and dropout
* ImageDataGenerator for augmentation
* Model evaluation with accuracy, precision, recall, F1-score, and ROC AUC
* Class weighting to address imbalanced data

### **THEORETICAL BACKGROUND**

**Convolutional Neural Networks (CNNs)**

* Designed for image-like data (e.g., spectrograms)
* Extract spatial and temporal features effectively

**Regularization and Stability**:

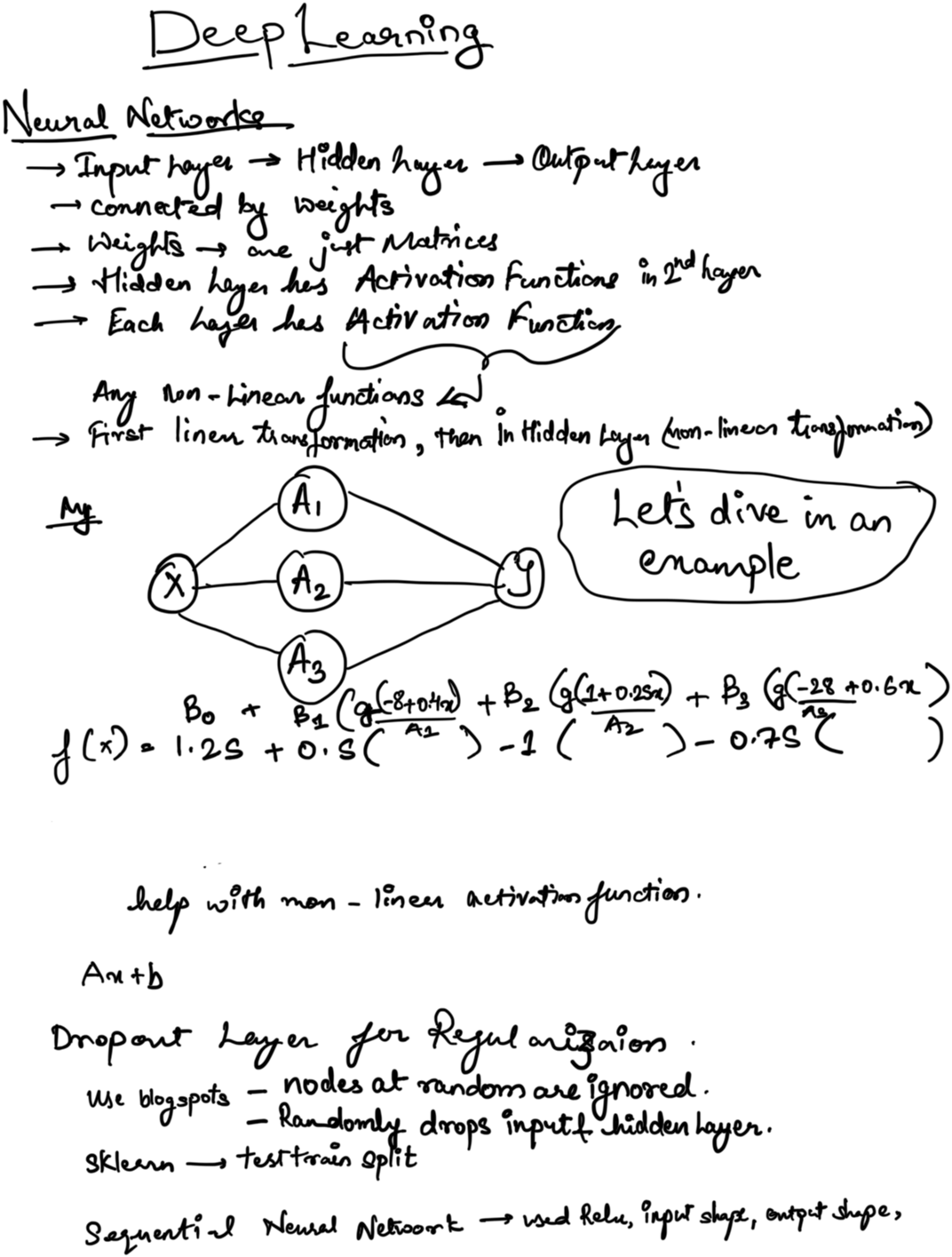
* **L2 Regularization**: Penalizes large weights to prevent overfitting
* **Batch Normalization**: Normalizes activations to stabilize training
* **Dropout**: Randomly disables neurons during training to improve generalization

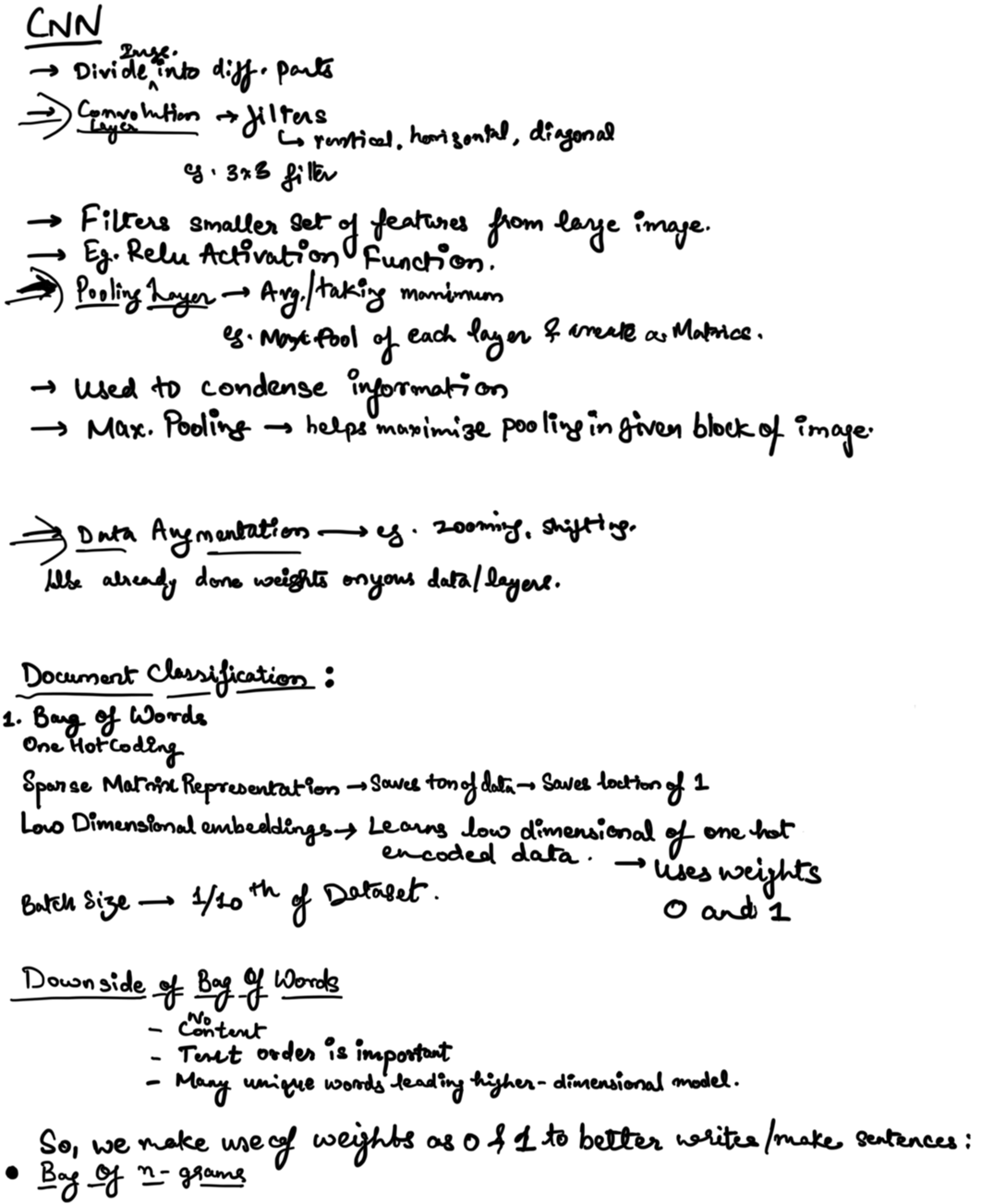
**Optimizers**:

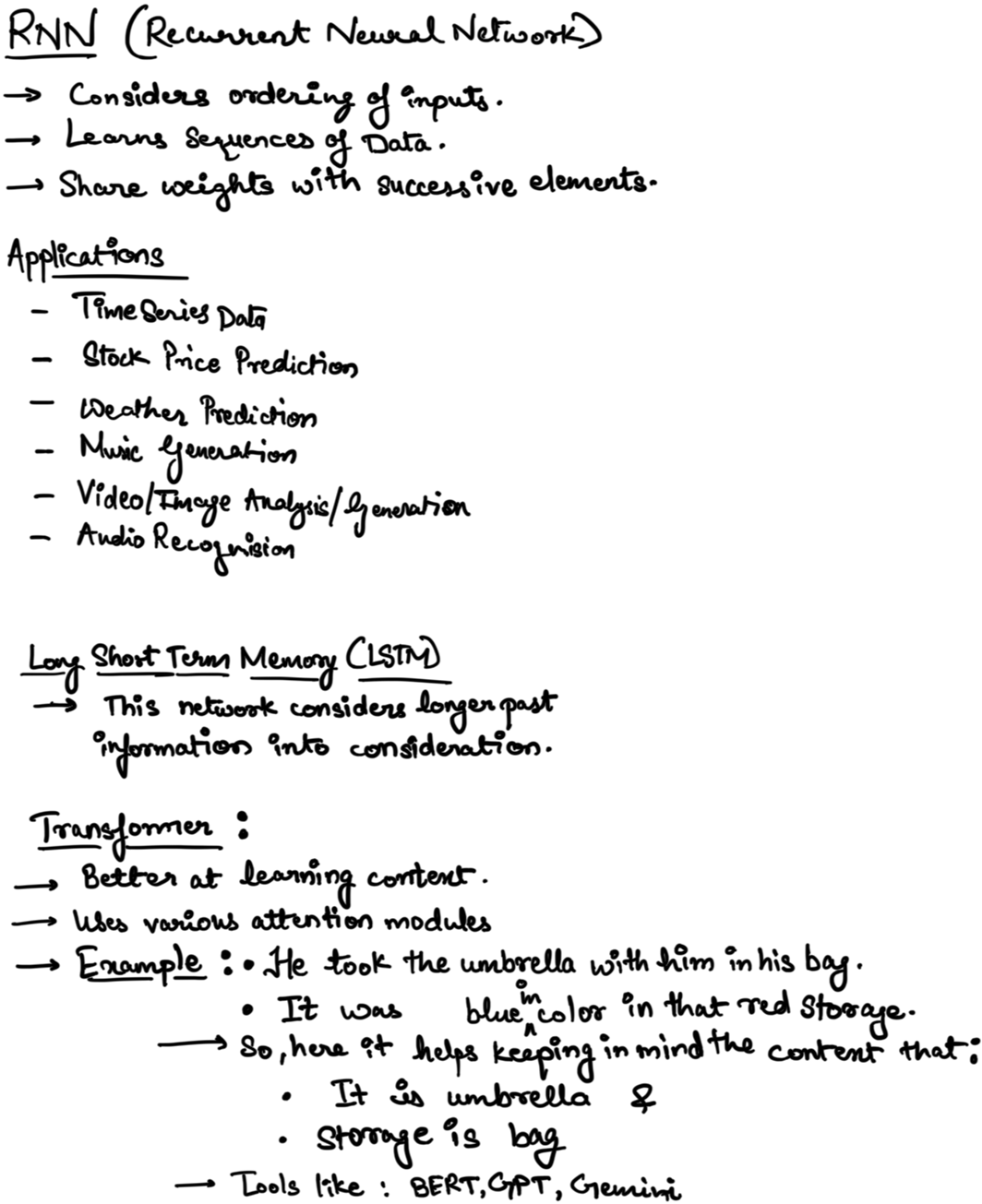
* **Adam**: Combines momentum and RMSprop for adaptive learning
* **RMSProp**: Suitable for handling noisy or sparse data with moving average updates

**Callbacks**:

* **EarlyStopping**: Stops training when validation loss stops improving
* **ReduceLROnPlateau**: Lowers learning rate when a monitored metric stalls







### **METHODOLOGY**

**Data Preparation**:

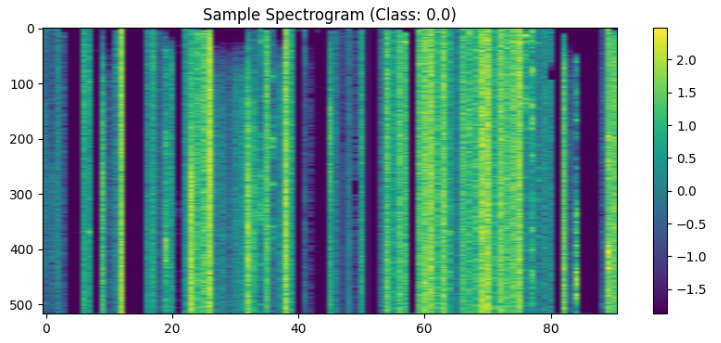
* Spectrograms read from .hdf5 format
* Cropped to consistent time length and normalized
* Data split: 76% training / 24% testing
* Augmentation: ImageDataGenerator with width\_shift=0.1, height\_shift=0.1, validation\_split=0.2

### **Binary Classification Model**

**Bird Species**:

* 'whcspa' (White-crowned Sparrow)
* 'rewbla' (Red-winged Blackbird)



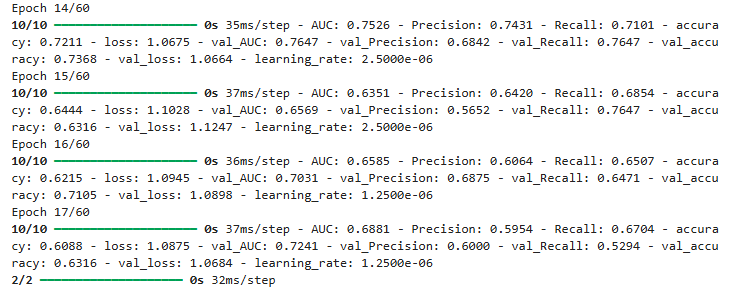


**Model Configurations**:

* **Adam Config**:
  + Filters: [4, 8]
  + Dropout: 0.5
  + Dense Units: 16
  + L2 Regularization: 0.01
  + Optimizer: Adam (LR = 1e-5)
* **RMSprop Config**:
  + Filters: [4, 8]
  + Dropout: 0.2
  + Dense Units: 8
  + L2 Regularization: 0.02
  + Optimizer: RMSProp (LR = 1e-5)

**Training Setup**:

* Epochs: 60
* Batch Size: 16
* EarlyStopping (patience=5)
* ReduceLROnPlateau (factor=0.5)

****

### **Multi-Class Classification Model**

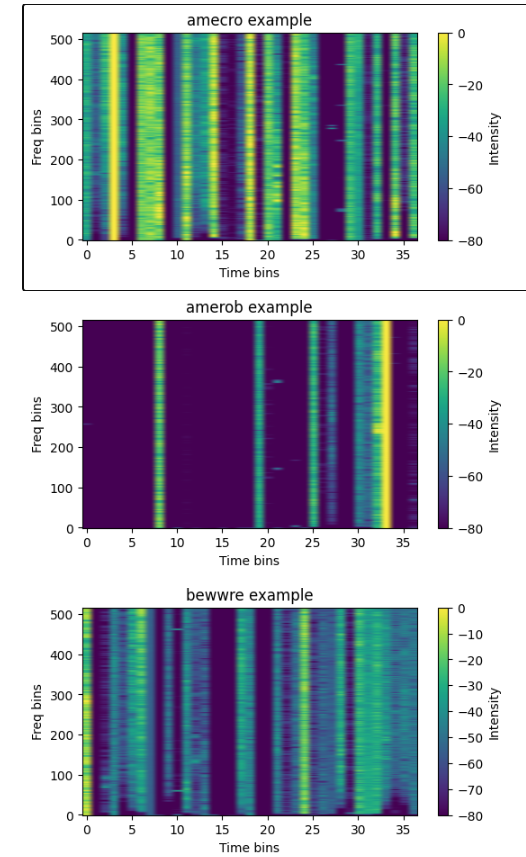
**All 12 bird species included**

**Configurations**:

* **regularized\_simple**:
  + Conv Layers: 2
  + Dense Units: 128
  + L2 Regularization: 0.001
  + Optimizer: Adam (LR = 1e-5)
* **deeper\_regularized**:
  + Conv Layers: 4
  + Dense Units: 256
  + L2 Regularization: 0.002
  + Optimizer: RMSProp (LR = 1e-5)

**Training Setup**:

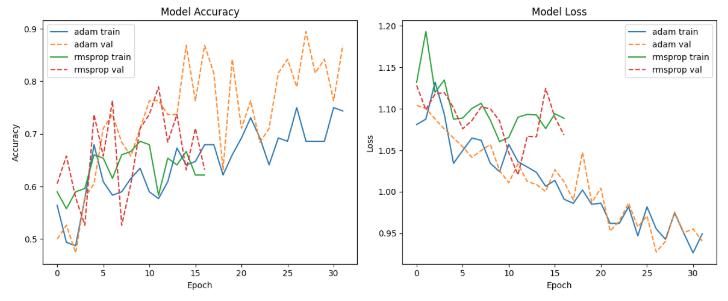
* Epochs: 120
* Batch Size: 32
* EarlyStopping (patience=10)
* ReduceLROnPlateau (factor=0.5)

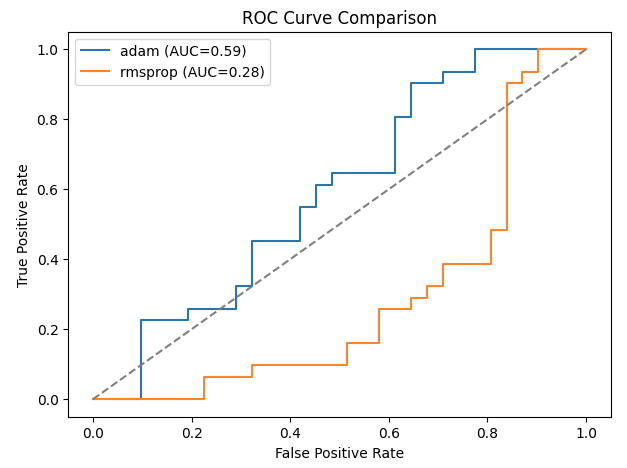


### **RESULTS AND COMPARISON**

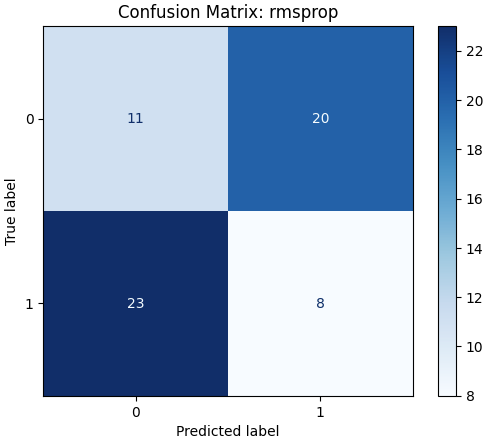
### **Table 1: Final Binary Classification Metrics**

| **Config** | **Accuracy** | **Precision** | **Recall** | **F1 Score** | **ROC AUC** | **Specificity** |
| --- | --- | --- | --- | --- | --- | --- |
| adam | 0.5968 | 0.5600 | 0.9032 | 0.6914 | **0.5921** | 0.2903 |
| rmsprop | 0.3065 | 0.2857 | 0.2581 | 0.2712 | 0.2810 | **0.3548** |

****

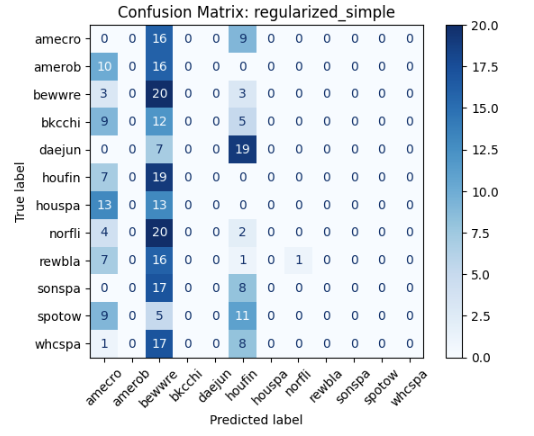
****

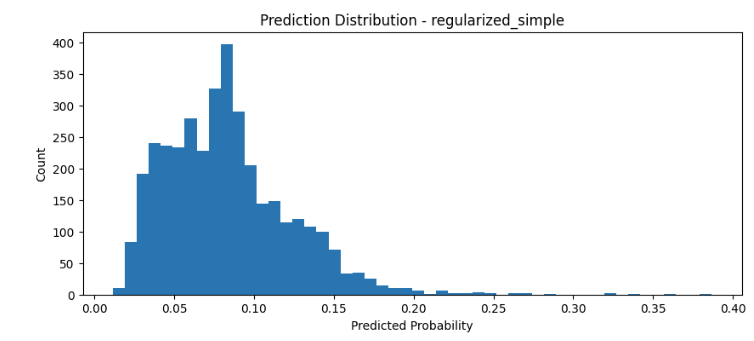
### 

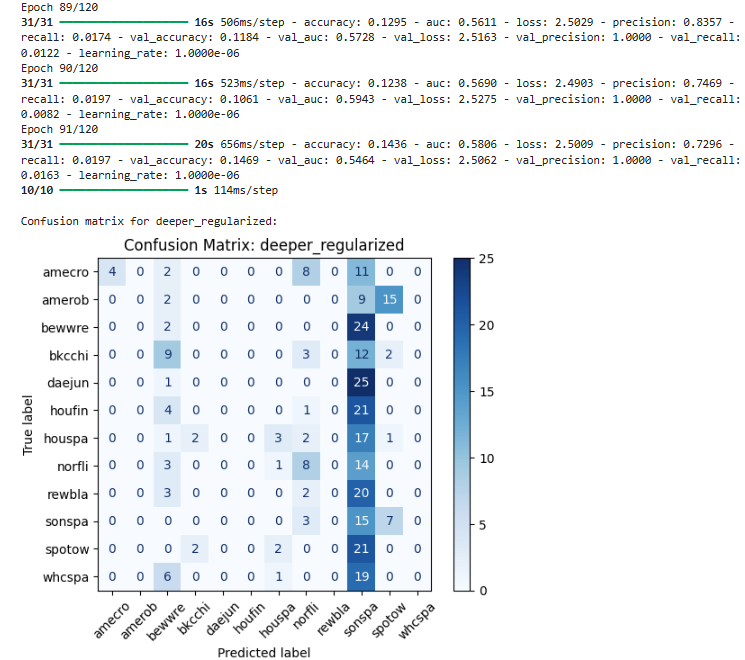


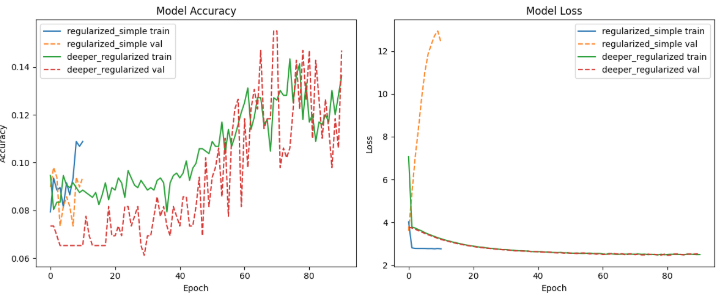
### **Table 2: Final Multiclass Classification Metrics**

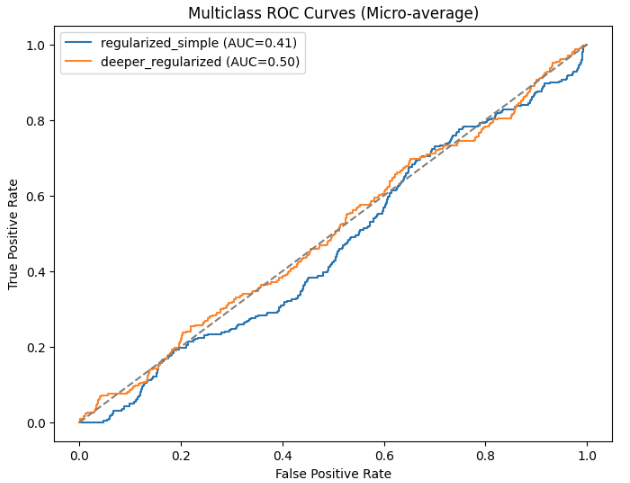
| **Config** | **Accuracy** | **Precision (Macro)** | **Recall (Macro)** | **F1 Score (Macro)** | **ROC AUC (Macro)** | **Training Time (min)** |
| --- | --- | --- | --- | --- | --- | --- |
| deeper\_regularized | **0.1039** | **0.1548** | **0.1050** | **0.0797** | **0.5044** | 124.73 |
| regularized\_simple | 0.0649 | 0.0094 | 0.0641 | 0.0163 | 0.4071 | **60.08** |





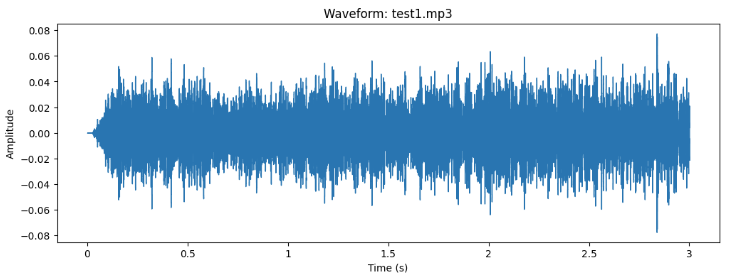


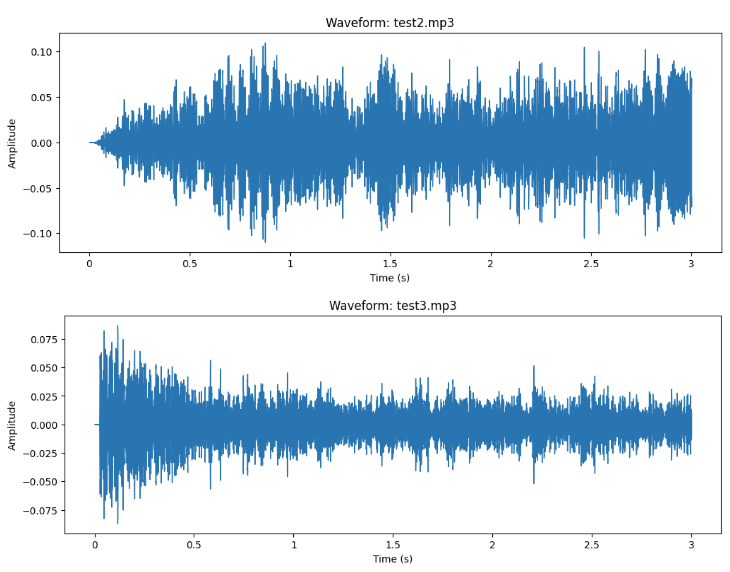


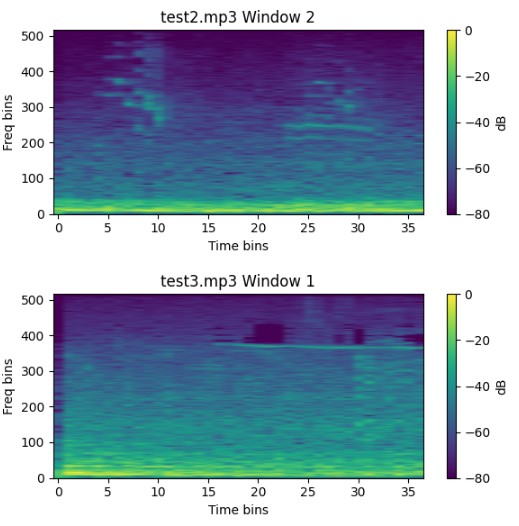
****

**Table 3: External Test Data Top-3 Predictions**

| **Clip** | **Top-1 Species** | **Top-1 Prob** | **Top-2 Species** | **Top-2 Prob** | **Top-3 Species** | **Top-3 Prob** |
| --- | --- | --- | --- | --- | --- | --- |
| test1.mp3 | houspa | **0.4578** | bkcchi | 0.3161 | norfli | 0.0773 |
| test2.mp3 | houspa | **0.2954** | bkcchi | 0.2479 | norfli | 0.1041 |
| test3.mp3 | houspa | **0.3678** | bkcchi | 0.2828 | norfli | 0.0947 |







### **VISUAL INSIGHTS**

**Spectrogram & Waveform Analysis**:

* Show time-frequency patterns of bird vocalizations.
* Variation among species is visible in frequency range and duration.

### **DISCUSSION**

**i. Training Time**

* Binary: Adam trained faster and more effectively than RMSprop.
* Binary: Adam outperformed RMSprop due to better generalization
* Class weights and augmentation helped balance learning.
* Multi-class: deeper\_regularized took significantly longer (~33 min).

**ii. Challenges**

* Initial overfitting (100% accuracy) resolved with regularization.
* Class imbalance (e.g., houspa dominating)
* Similar-sounding birds (e.g., bewwre, bkcchi) caused confusion
* Limited compute resources (no GPU) increased training time

**iii. Observations**

* Results vary across runs (despite seed) due to training dynamics.
* EarlyStopping helped reduce unnecessary epochs and saved time.

**iv. Why CNN?**

* Spectrograms are visual representations well-suited for CNN feature extraction.
* It can easily handled spectrogram in comparison to the other non-robust models.
* Helps in handling complex data.

**v. Alternative Approaches**:

* Use RNNs (e.g., LSTM, GRU) for temporal sequences
* Pretrained models like ResNet for better feature learning
* Lastly use of Transformers can be done which will ease the process as it retains/ remembers the pattern between the data.

**vi. Limitations**:

* In binary, there was limited data points/ sizes of data which at start affected showing a proper accuracy of 100%.
* Mixture of one or More species in an audio clip sometimes falsely predict the correct accuracy.
* External sound or background noise if any can alter the prediction or less the probability of detection.

**CONCLUSION**:

* This work demonstrates the feasibility of applying CNNs to classify bird species from spectrogram data.
* With appropriate tuning and preprocessing, even simple CNNs can effectively learn acoustic patterns.
* The model could be deployed in real-world bird monitoring systems, aiding ornithologists and conservationists.

**FUTURE SCOPE:**

* Expand and clean dataset
* Use transfer learning with pretrained models
* Use of Transformers to make it attentive and accurate at the same time.
* Explore RNNs to capture temporal data and its dependencies.
* Techniques to Handle environmental or external background noise. This will help to improve the robustness.

**REFERENCES / CITATIONS:**

[1] M. Abadi *et al*., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. [Online]. Available:<https://www.tensorflow.org/>

[2] F. Chollet *et al*., "Keras," 2015. [Online]. Available:<https://keras.io/>

[3] F. Pedregosa *et al*., "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[4] B. McFee *et al*., "librosa: Audio and music signal analysis in Python," in *Proc. 14th Python in Science Conf. (SciPy 2015)*, 2015, pp. 18–25. [Online]. Available:<https://librosa.org/>

[5] *Deep Learning 1*, Canvas Seattle University. [Online]. Available: <https://seattleu.instructure.com/courses/1621163/files/72015417?module_item_id=18402069>. [Accessed: 28/04/2025].

[6] *Deep Learning 2*, Canvas Seattle University. [Online]. Available: <https://seattleu.instructure.com/courses/1621163/files/72038418?module_item_id=18402075>. [Accessed: 05/05/2025].