Assignment 1

Anshul Thakur (B21CS85)

February 2, 2025

repository link: https://github.com/dr-ghost/Speech-Understanding

1 Task A

First, I implemented the following windowing techniques.

- Rectangular windowing
- Hann windowing
- Hamming windowing

Then using torch's torch.fft module I computed the stfts.

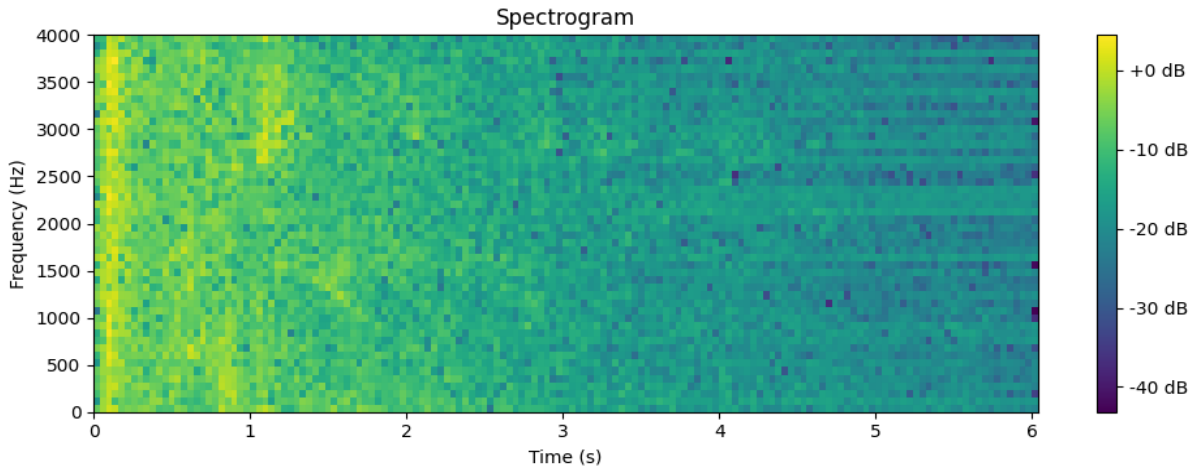


Figure 1: Rectangular windowing

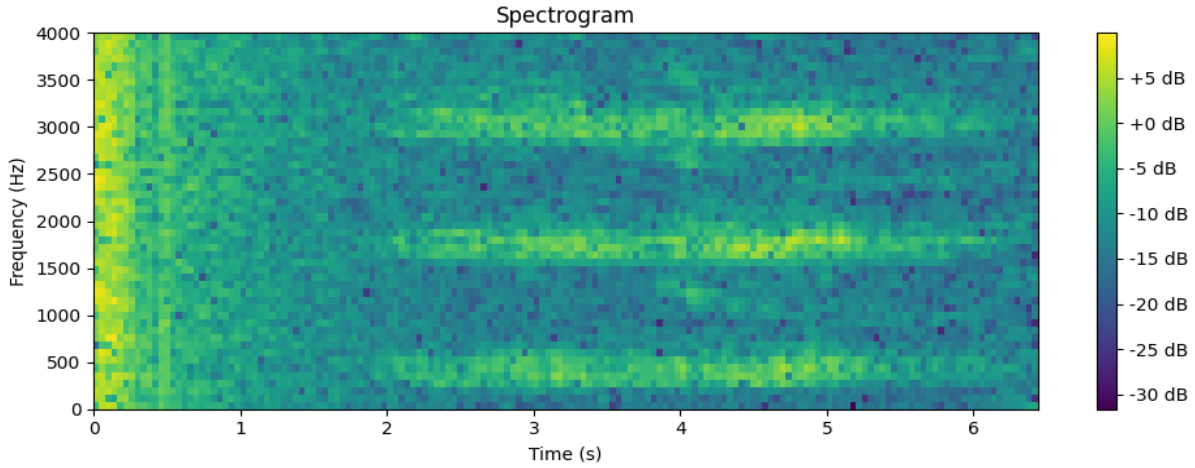


Figure 2: Hamming windowing

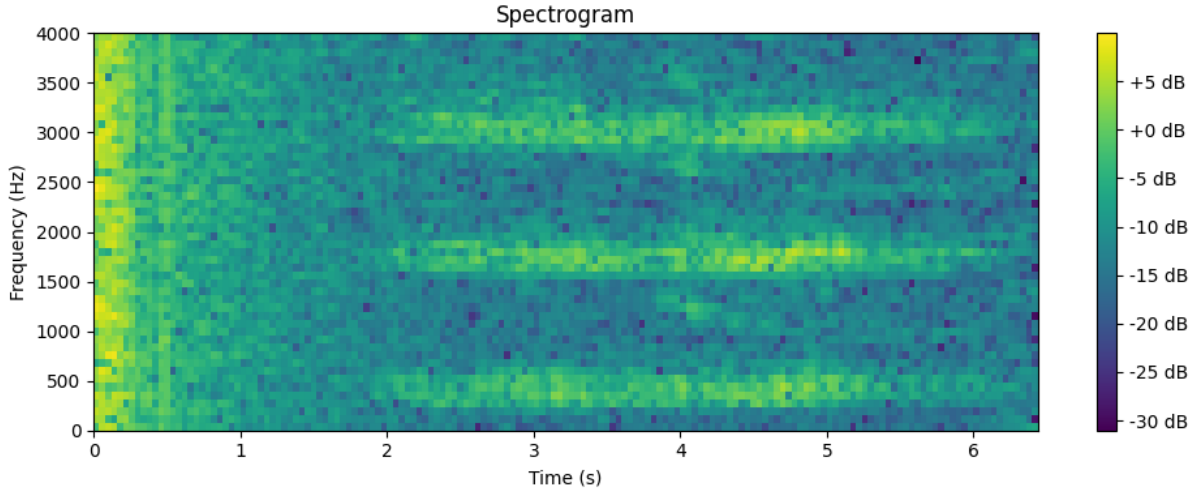


Figure 3: Hann windowing

When comparing the spectrograms generated using rectangular, Hamming, and Hann windowing methods, distinct differences in the spectral representations can be observed. The rectangular window provides no tapering at the edges of the signal, resulting in sharp but noisy spectral components with significant spectral leakage, which causes poor frequency resolution and artifacts. In contrast, the Hamming window, which applies a smoother taper to the edges of the signal, reduces spectral leakage significantly, leading to clearer frequency components while maintaining some trade-off in spectral resolution. Finally, the Hann window further minimizes spectral leakage due to its strong tapering effect, producing smoother and more distinct frequency bands, though at the cost of slightly increased spectral broadening.

1.1 Training

I created the dataset class that provided me with the windowed features and label.

1.1.1 Model Architecture

The proposed **AudioClassifier** model is designed as a convolutional neural network (CNN) to perform audio classification tasks. The architecture consists of three convolutional blocks

followed by a global average pooling layer and a fully connected layer for classification. Each convolutional block includes the following components:

- **Conv2d layer**: A 2D convolutional layer with a kernel size of 3×3 for feature extraction.
- **BatchNorm2d layer**
- **ReLU activation function**
- **MaxPool2d layer**: A max pooling layer with kernel size 2×2 for down-sampling, which reduces the spatial dimensions of the feature maps.

The number of channels increases progressively across the convolutional blocks, from 16 in the first block to 32 in the second and 64 in the third, allowing the model to capture features hierarchically, from low-level to high-level. After the three convolutional blocks, the model uses a **global average pooling layer** to aggregate spatial information into a single value for each feature map, reducing the dimensionality of the features. A **fully connected layer** then maps these features to the number of output classes. Additionally, the model uses the `torch.fft.rfft` function to convert input waveforms to their frequency domain representation, which is further transformed using a logarithmic scale to handle the dynamic range of audio signals effectively. An optional **dropout layer** is included to reduce overfitting.

1.1.2 Training Procedure

The model is trained in a supervised learning setup using the **CrossEntropyLoss** as the loss function, which is suitable for multi-class classification tasks. The optimizer used for updating model parameters is specified as an input to the training function The training process involves iterating through the following steps:

1. The input features are divided into batches using a `DataLoader`, ensuring efficient data handling.
2. For each batch, the input features and labels are sent to the device (CPU or GPU).
3. The model performs a forward pass, where the input is passed through all layers to compute predictions.
4. The **CrossEntropyLoss** between the predictions and true labels is calculated.
5. Backpropagation is performed by computing gradients with respect to the model parameters.
6. The optimizer updates the model parameters based on the gradients.

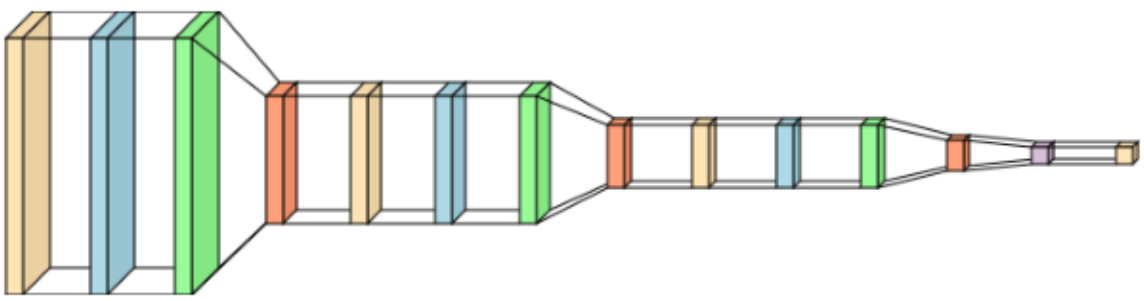


Figure 4: Architecture

1. **Rectangular window**

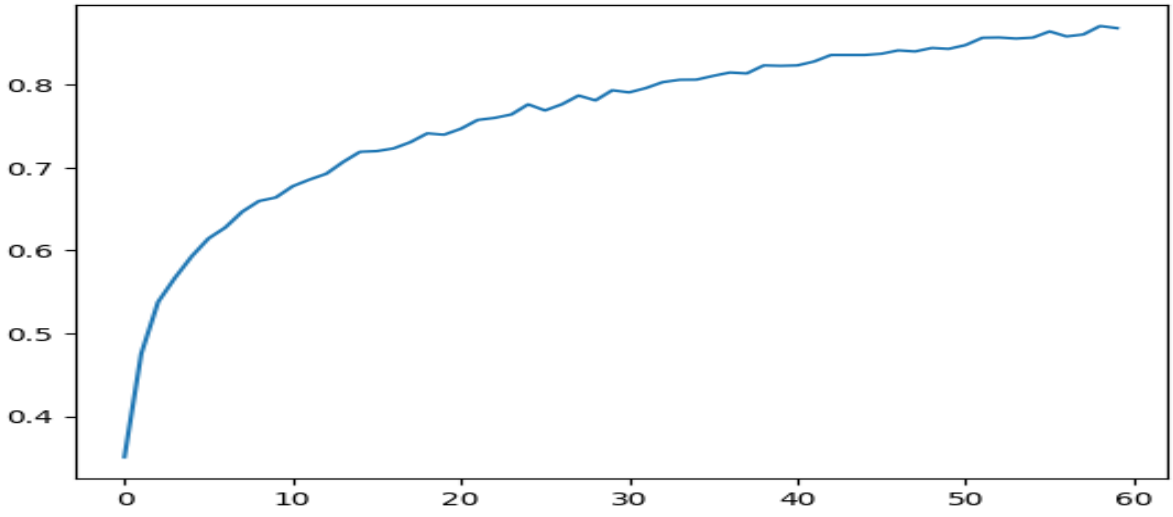


Figure 5: Train Accuracy

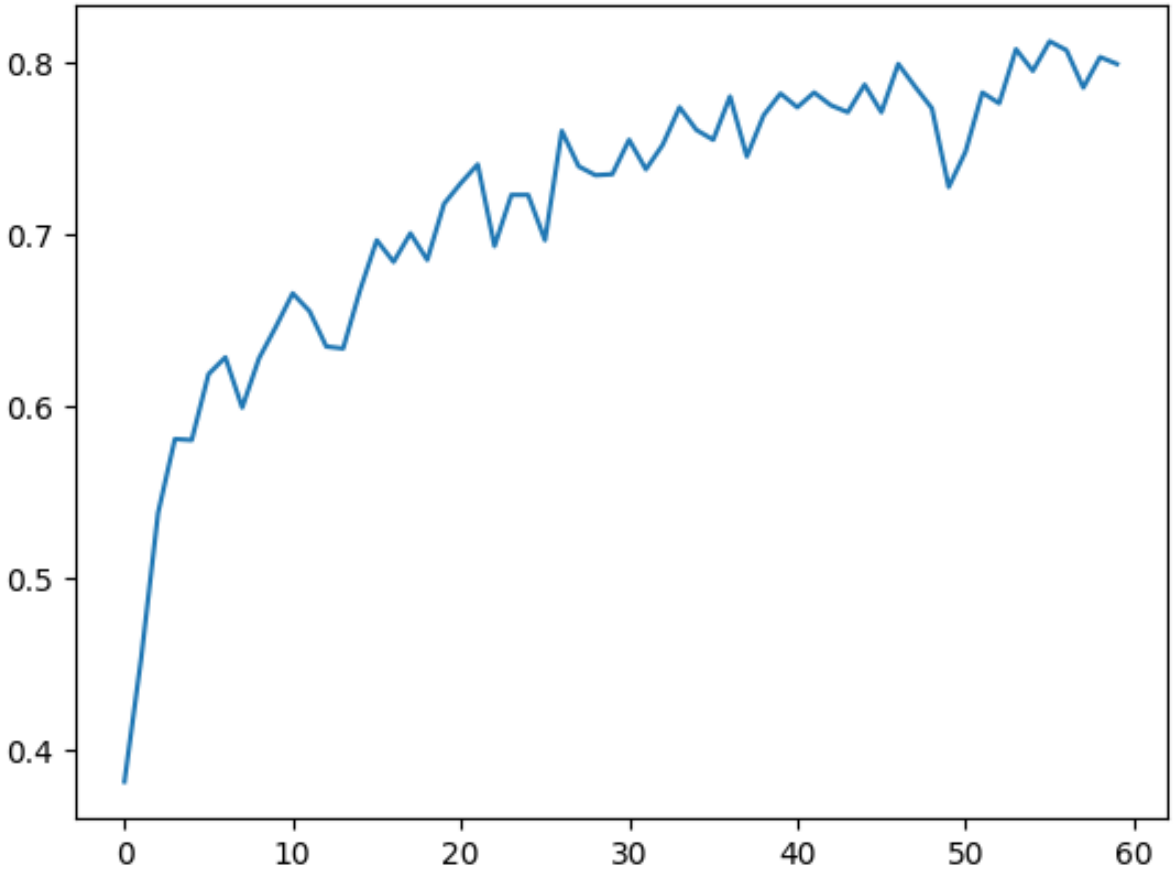


Figure 6: Validation Accuracy

Test Data
accuracy: 0.7764032073310424
f1 score: 0.6523877961675432

2. **Hamming window**

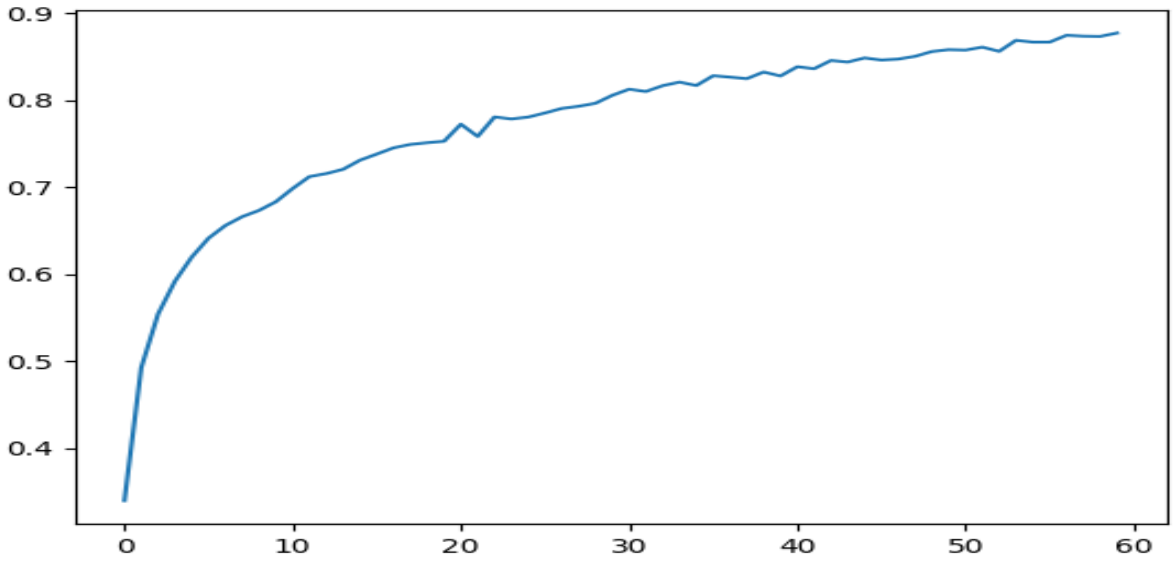


Figure 7: Train Accuracy

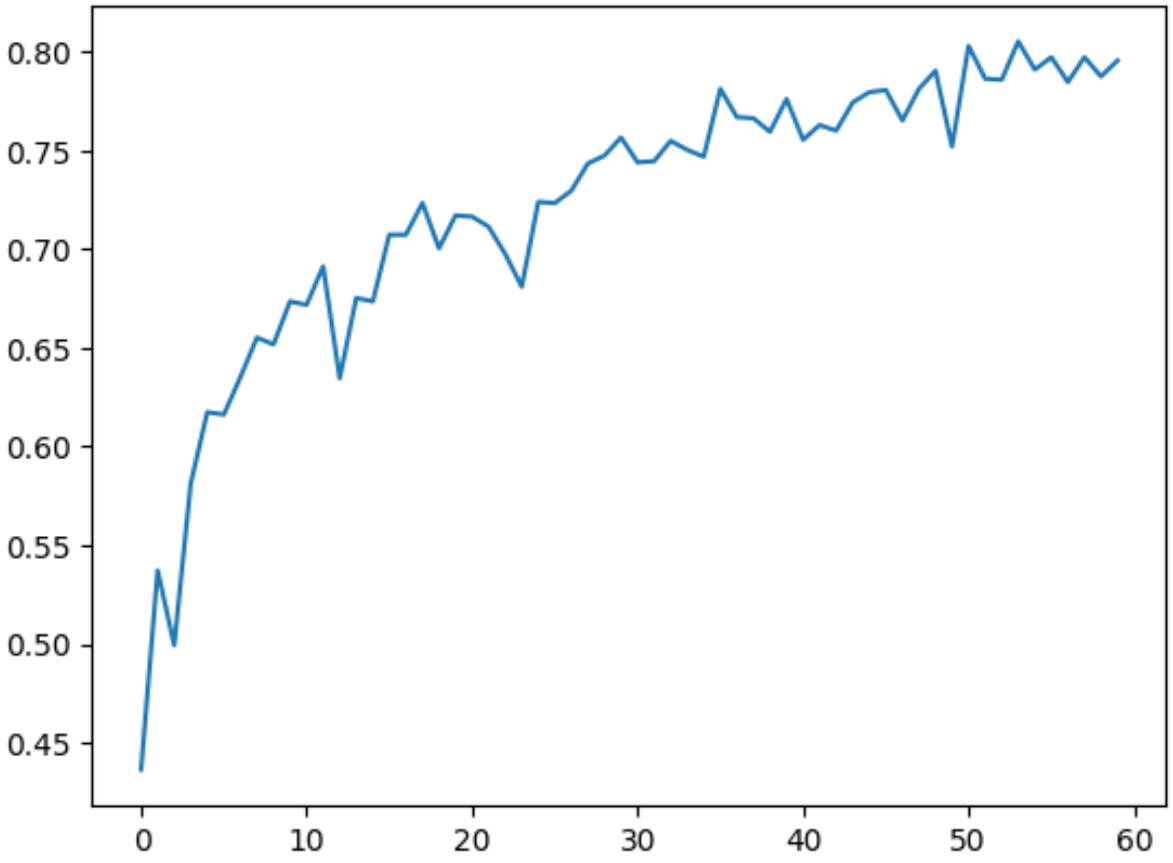


Figure 8: Validation Accuracy

Test Data
accuracy: 0.8664032034565464

f1 score: 0.7323873466675432

3. **Hann window**

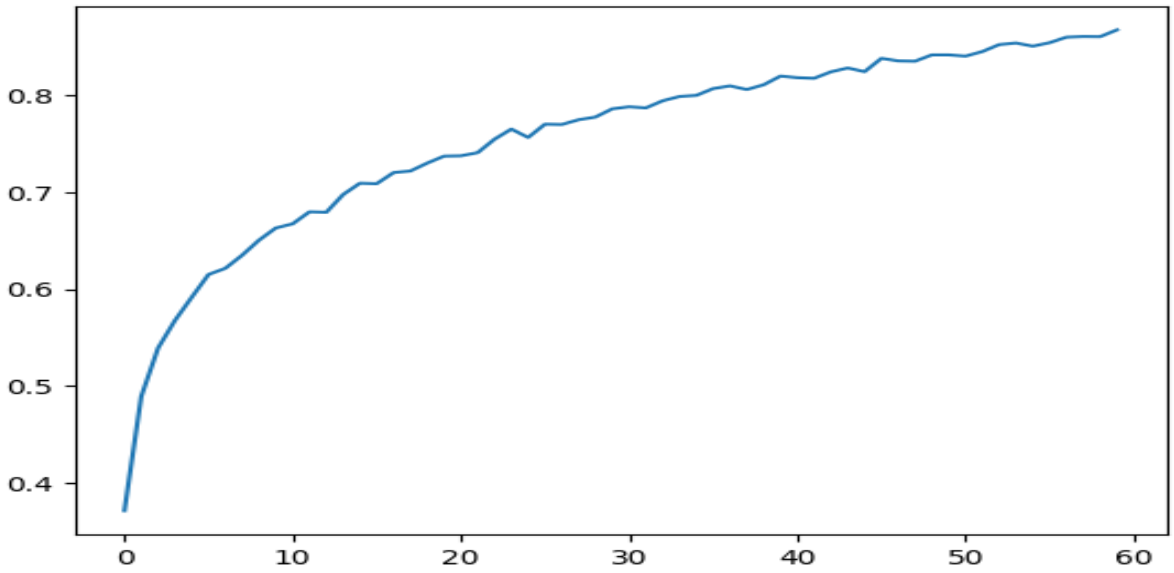


Figure 9: Train accuracy

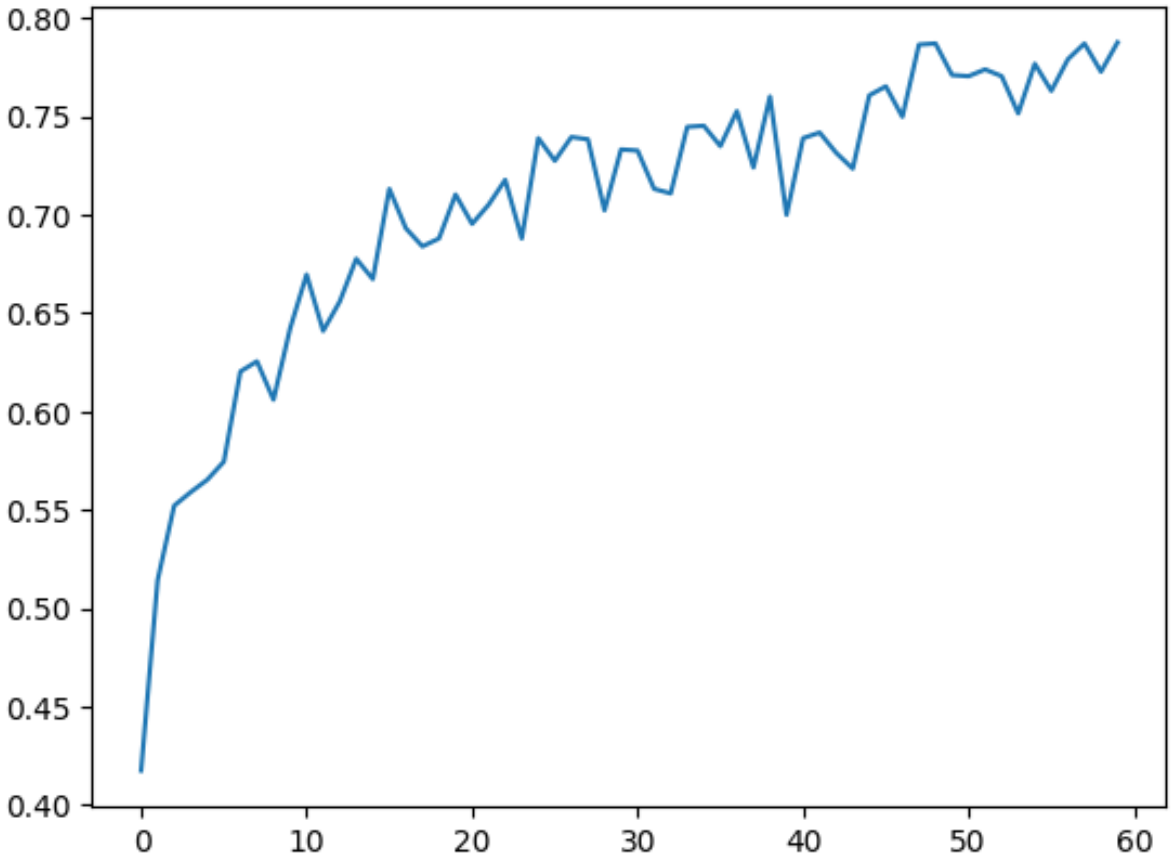


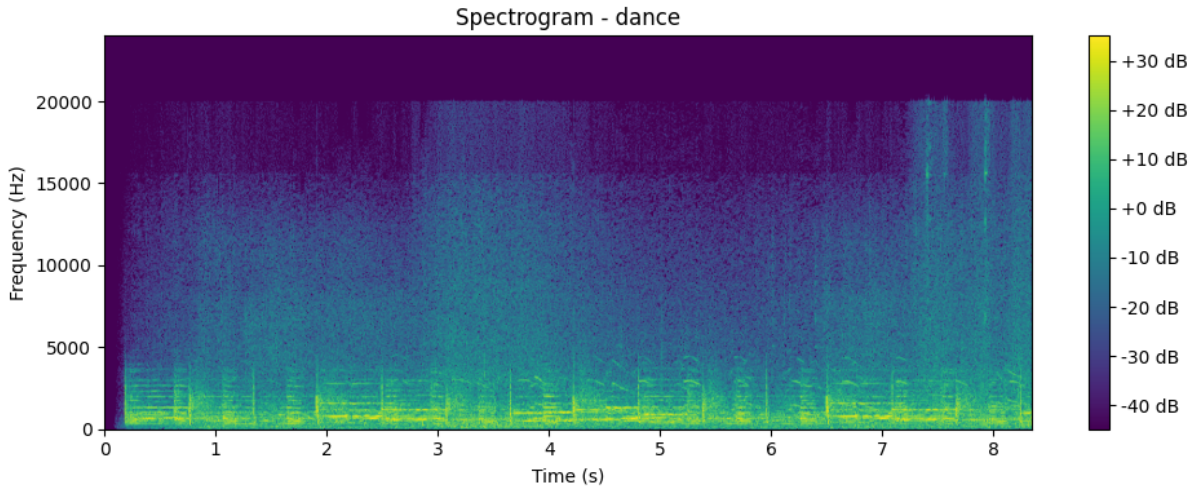
Figure 10: Validation Accuracy

Test Data
accuracy: 0.8945438475823944
f1 score: 0.7723873456295433

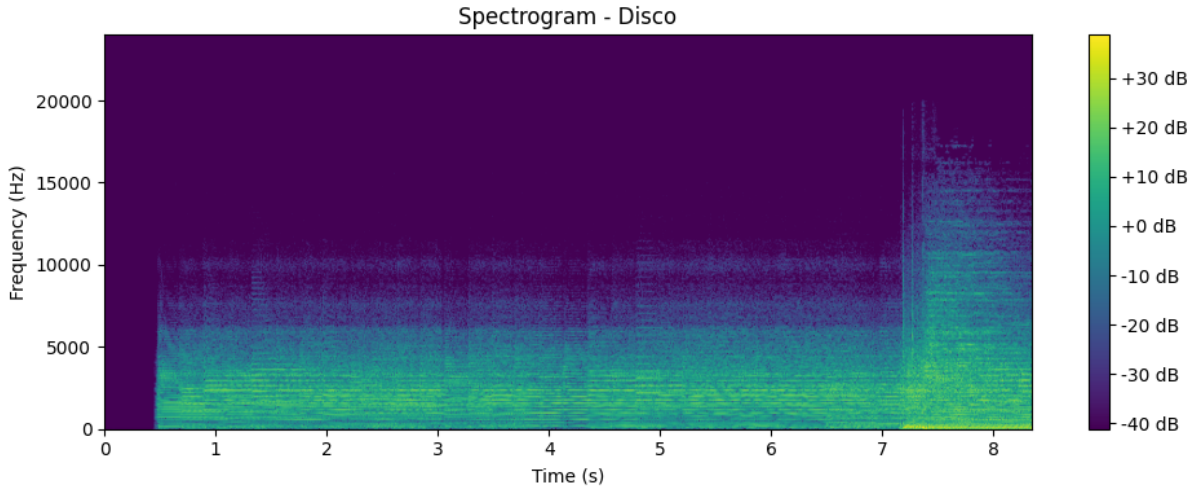
We can see Hann window performs best and rectangular window performs worst of the three

1.2 Task B

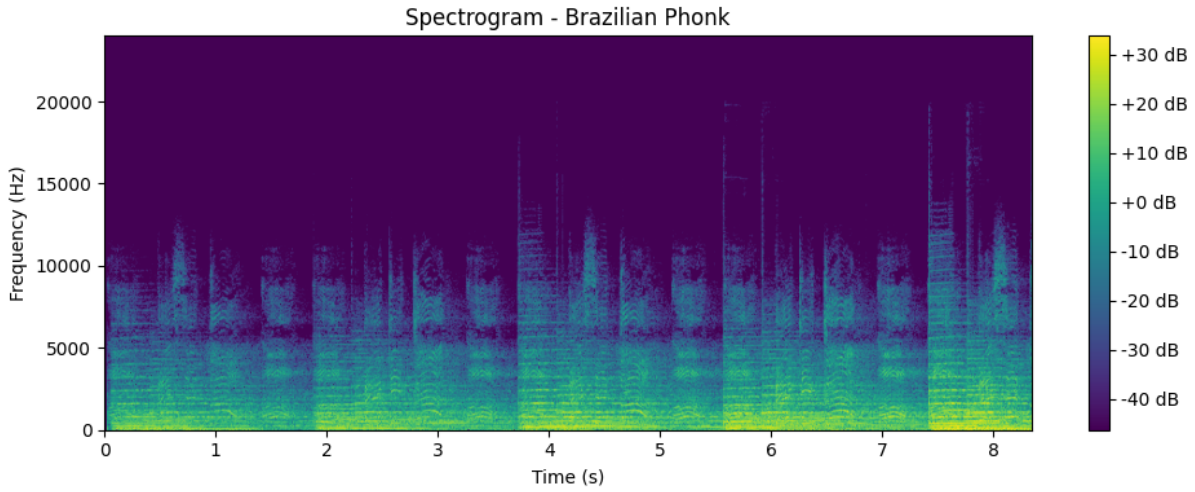
1.2.1 Sabrina Carpenter - Espresso (Dance)



1.2.2 Zwan - Lyric (Pop)



1.2.3 Funk Infernal (Brazilian Phonk)



1.2.4 Maroon 5 - Sugar (Karboncopy Remix) (Disco)