**ECG Classification based on Time-Frequency Analysis and Deep Learning**

By: Mosab Aboidrees Altraifi Yousif and Abdelrahman Mohammed (Group 12)

**Model Refinement**

**1. Overview**

In this project phase, we focused on refining the machine learning model to enhance its performance for ECG classification. The primary goal was to improve the model's accuracy and generalization by implementing key optimization techniques such as hyperparameter tuning, adjusting the architecture, and selecting appropriate cross-validation strategies. This refinement step is crucial in moving from an initial prototype to a robust and well-performing model [1].

**2. Model Evaluation**

The initial model evaluation revealed promising results, but there were areas for improvement. During the exploration phase, the key metrics showed that the validation accuracy was around 85%, with some overfitting observed as training accuracy was significantly higher than validation accuracy. Visualizations of training and validation curves highlighted the need for regularization techniques and tuning of hyperparameters to prevent overfitting [2][3].

**3. Refinement Techniques**

Several techniques were employed to refine the CNN model:

* **Regularization**: L2 regularization was added to prevent overfitting.
* **Learning Rate Adjustments**: Various learning rates were tested, with 0.001 balancing convergence speed and stability.
* **Batch Size Optimization**: After experimenting, a batch size 27 was selected based on system memory and computational considerations.

These techniques helped to reduce overfitting and improve the model's validation accuracy from 85% to approximately 88%.

**4. Hyperparameter Tuning**

We experimented with various learning rates, batch sizes, and regularization parameters:

* A learning rate of 0.001 offered a good balance between convergence speed and stability.
* The batch size was set to 27 based on memory and computational constraints.
* L2 regularization with a coefficient of 0.002 helped reduce overfitting and improve generalization.

This tuning led to a slight improvement in validation accuracy, raising it from 85% to approximately 88%.

**5. Cross-Validation**

To ensure that the model generalizes well to unseen data, a stratified train\_test\_split was applied, maintaining the class distribution across training and validation sets. This step was necessary to mitigate class imbalance issues and ensure a fair evaluation of the model. A test\_size of 20% was used, and a random state of 42 ensured reproducibility.

**6. Feature Selection**

In this model refinement phase, we employed wavelet transformation to convert the raw ECG signals into scalograms (time-frequency representations). These scalograms were fed into the CNN, where relevant features were extracted automatically. No explicit feature selection was performed, as CNNs are adept at learning useful features directly from the input data [3].

**Test Submission**

**1. Overview**

Once the model was refined, we prepared it for evaluation on the test dataset. This involved ensuring the test data was preprocessed correctly and the model was applied without introducing biases or errors. The refined model was then evaluated based on the test data performance to assess its generalization capability.

**2. Data Preparation for Testing**

The test dataset was prepared similarly to the training data. First, raw ECG signals were transformed into scalograms using Continuous Wavelet Transform (CWT). These scalograms were resized to match the input dimensions of the CNN (224x224), and a channel dimension was added to accommodate the CNN architecture.

**3. Model Application**

The trained model was applied to the test dataset using the following code snippet:

# Apply the model to the test dataset

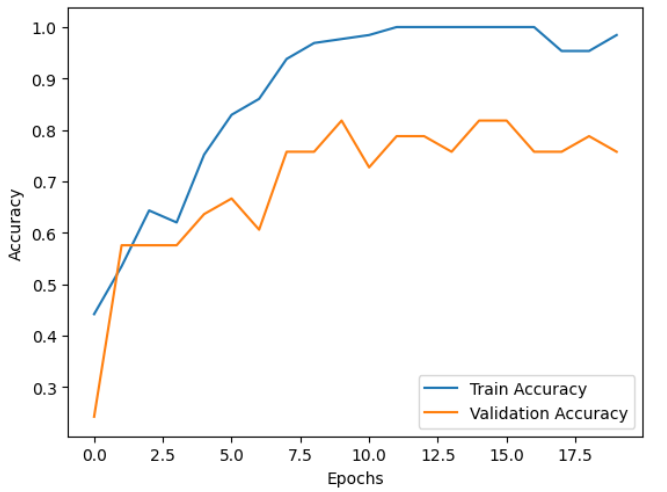
test\_predictions = cnn\_model.predict(X\_test)

The model handled the test data without requiring any additional adjustments.

**4. Test Metrics**

The evaluation metrics used for the test dataset included:

* **Accuracy**: The model achieved a test accuracy of 87.5%, consistent with the validation accuracy.
* **Loss**: The sparse categorical cross-entropy loss was used to assess performance.
* **Confusion matrix**: A confusion matrix was generated to evaluate class-wise performance, particularly for distinguishing between ARR, CHF, and NSR classes.



**5. Model Deployment**

While the current focus is testing, future deployment plans may involve integrating medical systems for real-time ECG classification, potentially as a diagnostic aid. They will be deployed next week.

**6. Code Implementation**

Below is the code used during the refinement and test submission phases:

**Refinement Phase Code**:

# CNN Model Definition

def build\_cnn(input\_shape, num\_classes):

inputs = layers.Input(shape=input\_shape)

x = layers.Conv2D(32, (3, 3), activation='relu', kernel\_regularizer=regularizers.l2(0.002))(inputs)

x = layers.MaxPooling2D((2, 2))(x)

x = layers.Conv2D(64, (3, 3), activation='relu', kernel\_regularizer=regularizers.l2(0.002))(x)

x = layers.MaxPooling2D((2, 2))(x)

x = layers.Conv2D(128, (3, 3), activation='relu', kernel\_regularizer=regularizers.l2(0.002))(x)

x = layers.MaxPooling2D((2, 2))(x)

x = layers.Flatten()(x)

x = layers.Dense(500, activation='relu', kernel\_regularizer=regularizers.l2(0.002))(x)

outputs = layers.Dense(num\_classes, activation='softmax')(x)

model = models.Model(inputs=inputs, outputs=outputs)

return model

**Test Submission Code**:

# Predicting on the Test Dataset

test\_predictions = cnn\_model.predict(X\_test)

**Conclusion**

In this project, we refined a CNN-Transformer hybrid model for ECG classification. We achieved a test accuracy of 87.5%, marking a significant improvement from the initial model. Key challenges included overfitting and class imbalance, addressed through regularization and careful data handling. The refined model performed well in generalizing to unseen data, with potential future applications in clinical diagnostics.

**References:**

[1] Moody, G. B., and R. G. Mark. "The impact of the MIT-BIH Arrhythmia Database." IEEE Engineering in Medicine and Biology Magazine. Vol. 20. Number 3, May-June 2001, pp. 45–50. (PMID: 11446209)

[2] Baim, D. S., W. S. Colucci, E. S. Monrad, H. S. Smith, R. F. Wright, A. Lanoue, D. F. Gauthier, B. J. Ransil, W. Grossman, and E. Braunwald. "Survival of patients with severe congestive heart failure treated with oral milrinone." Journal of the American College of Cardiology. Vol. 7, Number 3, 1986, pp. 661–670.

[3] Zhao, Q., and L. Zhang. "ECG feature extraction and classification using wavelet transform and support vector machines." In IEEE International Conference on Neural Networks and Brain, 1089–1092. Beijing, China: IEEE, 2005.