

Model Development Phase Template

Date	21 June 2025
Project Title	Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

```
def build_shallow_cnn():  
    model = Sequential([  
        Conv2D(16, (3,3), activation='relu', input_shape=(128,128,3)),  
        MaxPooling2D(2,2),  
        Flatten(),  
        Dense(64, activation='relu'),  
        Dropout(0.3),  
        Dense(6, activation='softmax')  
    ])  
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])  
    return model
```

```
def build_deep_cnn():
    model = Sequential([
        Conv2D(32, (3,3), activation='relu', input_shape=(128,128,3)),
        MaxPooling2D(2,2),
        Conv2D(64, (3,3), activation='relu'),
        MaxPooling2D(2,2),
        Conv2D(128, (3,3), activation='relu'),
        MaxPooling2D(2,2),
        Flatten(),
        Dense(256, activation='relu'),
        Dropout(0.5),
        Dense(6, activation='softmax')
    ])
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
    return model
```

```
def build_cnn_lstm():
    model = Sequential([
        Conv2D(32, (3,3), activation='relu', input_shape=(128,128,3)),
        MaxPooling2D(2,2),
        Conv2D(64, (3,3), activation='relu'),
        MaxPooling2D(2,2),
        Flatten(),
        Reshape((1, -1)),
        LSTM(64, return_sequences=False),
        Dense(128, activation='relu'),
        Dropout(0.5),
        Dense(6, activation='softmax')
    ])
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
    return model
```

```
[3] class Attention(Layer):
    def __init__(self, **kwargs):
        super(Attention, self).__init__(**kwargs)
        self.att_dense = None

    def build(self, input_shape):
        self.att_dense = Dense(input_shape[-1], activation='softmax')
        super(Attention, self).build(input_shape)

    def call(self, inputs):
        attention_probs = self.att_dense(inputs)
        return Multiply()([inputs, attention_probs])

def build_cnn_gru_attention():
    input_layer = Input(shape=(128,128,3))
    x = Conv2D(32, (3,3), activation='relu')(input_layer)
    x = MaxPooling2D(2,2)(x)
    x = Conv2D(64, (3,3), activation='relu')(x)
    x = MaxPooling2D(2,2)(x)
    x = Flatten()(x)
    x = Reshape((1, -1))(x)
    x = GRU(64, return_sequences=True)(x)
    x = Attention()(x)
    x = Flatten()(x)
    x = Dense(128, activation='relu')(x)
    x = Dropout(0.5)(x)
    output = Dense(6, activation='softmax')(x)
    model = Model(inputs=input_layer, outputs=output)
    model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
    return model
```

Model Validation and Evaluation Report:

MODEL	CLASSIFICATION REPORT					F1 SCORE	CONFUSION MATRIX
Shallow CNN	Shallow CNN Classification Report:					67%	Shallow CNN Confusion Matrix: [[55 6 112 160 8 0] [0 1985 116 4 72 2] [0 70 1014 132 273 14] [24 52 76 514 248 1] [0 242 99 1 1303 0] [0 18 9 16 1 198]]
	precision	recall	f1-score	support			
	Left Bundle Branch Block	0.70	0.16	0.26	341		
	Normal	0.84	0.91	0.87	2179		
	Premature Atrial Contraction	0.71	0.67	0.69	1503		
	Premature Ventricular Contractions	0.62	0.56	0.59	915		
	Right Bundle Branch Block	0.68	0.79	0.73	1645		
	Ventricular Fibrillation	0.92	0.82	0.87	242		
	accuracy			0.74	6825		
	macro avg	0.75	0.65	0.67	6825		
weighted avg	0.74	0.74	0.73	6825			
Deep CNN	Deep CNN Classification Report:					91%	Deep CNN Confusion Matrix: [[339 0 0 1 1 0] [2 2064 84 10 17 2] [0 91 1376 27 7 2] [22 176 4 677 36 0] [1 203 19 0 1422 0] [0 9 1 5 0 227]]
	precision	recall	f1-score	support			
	Left Bundle Branch Block	0.93	0.99	0.96	341		
	Normal	0.81	0.95	0.87	2179		
	Premature Atrial Contraction	0.93	0.92	0.92	1503		
	Premature Ventricular Contractions	0.94	0.74	0.83	915		
	Right Bundle Branch Block	0.96	0.86	0.91	1645		
	Ventricular Fibrillation	0.98	0.94	0.96	242		
	accuracy			0.89	6825		
	macro avg	0.93	0.90	0.91	6825		
weighted avg	0.90	0.89	0.89	6825			
CNN with LSTM	CNN + LSTM Classification Report:					80%	CNN + LSTM Confusion Matrix: [[0 341 0 0 0 0] [0 2179 0 0 0 0] [0 1503 0 0 0 0] [0 915 0 0 0 0] [0 1645 0 0 0 0] [0 242 0 0 0 0]]
	precision	recall	f1-score	support			
	Left Bundle Branch Block	0.00	0.00	0.00	341		
	Normal	0.32	1.00	0.48	2179		
	Premature Atrial Contraction	0.00	0.00	0.00	1503		
	Premature Ventricular Contractions	0.00	0.00	0.00	915		
	Right Bundle Branch Block	0.00	0.00	0.00	1645		
	Ventricular Fibrillation	0.00	0.00	0.00	242		
	accuracy			0.32	6825		
	macro avg	0.05	0.17	0.08	6825		
weighted avg	0.10	0.32	0.15	6825			
CNN with GRU	CNN + GRU + Attention Classification Report:					60%	CNN + GRU + Attention Confusion Matrix [[0 341 0 0 0 0] [0 2179 0 0 0 0] [0 1503 0 0 0 0] [0 915 0 0 0 0] [0 1645 0 0 0 0] [0 242 0 0 0 0]]
	precision	recall	f1-score	support			
	Left Bundle Branch Block	0.00	0.00	0.00	341		
	Normal	0.32	1.00	0.48	2179		
	Premature Atrial Contraction	0.00	0.00	0.00	1503		
	Premature Ventricular Contractions	0.00	0.00	0.00	915		
	Right Bundle Branch Block	0.00	0.00	0.00	1645		
	Ventricular Fibrillation	0.00	0.00	0.00	242		
	accuracy			0.32	6825		
	macro avg	0.05	0.17	0.08	6825		
weighted avg	0.10	0.32	0.15	6825			

