ABSTRACT

LI, YIXIN. Cyber-Physical System in Healthcare: Real-time Cardiovascular Disease Detection through Low-Power Embedded Devices. (Under the direction of Zhishan Guo).

Real-time processing of electrocardiogram (ECG) signals on wearable devices is challenging due to heart rate variability, limited computational resources, and the necessity for immediate detection of cardiovascular disease (CVD). Traditional methods often struggle to balance diagnostic accuracy with computational efficiency under these constraints. To address these challenges, we propose a novel convolutional neural network architecture enhanced with Residual Blocks, Squeeze-and-Excitation layers, and a Global Attention mechanism, specifically designed for CVD detection. Additionally, we adapt an Adaptive Model Selection (AMS) that dynamically adjusts the model complexity in real-time based on the current heart rate to balance computational loads and system accuracy, maintaining real-time constraints. Integrated within a Cyber-Physical System, this model ensures seamless interaction between sensing, computation, and actuation for timely medical interventions. We evaluated the performance of the model using a varying number of heartbeats as input to optimize the diagnostic accuracy. Extensive evaluations using the PhysioNet Database on Raspberry Pi 4 demonstrate that our AMS model achieves a high diagnostic accuracy of 91.5% with two ECG cycles, maintaining a low inference time of 1.33 ms per sample to meet real-time processing constraints. This approach improves the efficiency and reliability of wearable ECG monitoring systems, contributing to better continuous cardiac care.

Cyber-Physical System in Healthcare: Real-time Cardiovascular Disease Detection through Low-Power Embedded Devices

by
Yixin Li

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Computer Science

Raleigh, North Carolina
2025

APPROVED BY:

Ning Sui	Wujie Wen
Chenhan Xu	Huining Li

Zhishan Guo
Chair of Advisory Committee

BIOGRAPHY

The author was born in Wuhan, China, and spent his early years there as a dedicated student athlete. Motivated by a desire to further his education, he moved to the United States and enrolled at North Carolina State University (NCSU), initially majoring in Sports Management. Despite facing significant language challenges upon his arrival, he remained determined to explore new opportunities, particularly in the field of technology. During his sophomore year, he took a bold step by transferring from the Sports Management program to the Computer Science department, becoming the only student in his cohort to do so.

After earning his bachelor's degree, he began working as a full-time engineer at a biotechnology company. However, his commitment to learning and growth led him to pursue a Ph.D. Over two years of focused research, he initiated a project in collaboration with UNC Hospital to develop artificial intelligence methods for detecting cardiovascular disease (CVD). The tragic loss of a close friend at the age of 22 due to heart failure during exercise deeply influenced his work, underscoring the urgent need for accurate, real-time CVD detection technology.

Eager to continue addressing real-world challenges and to further build his industrial expertise, he recently accepted a role as a Product Manager at Lenovo in the highly competitive Global Future Leaders Program. He remains passionate about his work, viewing every challenge as an opportunity to learn and make a meaningful impact.

TABLE OF CONTENTS

List of Tables . . . . . **v**

List of Figures . . . . . **vi**

Chapter 1 INTRODUCTION **1**
1.1 Definitions 1
1.2 Motivations 2
1.3 Background 4
1.4 Problem Statement 5
1.4.1 Problem Formulation 6
1.4.2 Objective 7

Chapter 2 Model Design **8**
2.1 Approach 1: Self-Attention CNN-LSTM 8
2.1.1 ECG Pattern Recognition 8
2.1.2 Model Design 10
2.2 Approach 2: Attention-based SE-ResNet 11
2.2.1 Model Workflow 11
2.2.2 Residual Blocks with Squeeze-and-Excitation Layers 12
2.2.3 Global Attention Mechanism 13
2.2.4 Output Layer 14

Chapter 3 Framework **15**
3.1 System Design 15
3.1.1 Shifted Window Approach 15
3.1.2 Adaptive Model Selection 17
3.1.3 Anytime Model with Parameter Sharing 18
3.2 Schedulability Analysis for Real-time Processing 20
3.2.1 System Model 20
3.2.2 Scheduling Algorithm 21
3.2.3 Schedulability Analysis 21

Chapter 4 Experiments And Results **23**
4.1 Experiments and Results 23
4.1.1 Experimental Implementation 23
4.1.2 Experimental Results 26

Chapter 5 Discussion **28**
5.1 Discussion 28
5.1.1 Model Performance Comparison 28
5.1.2 AMS and the Anytime Model 29
5.1.3 Practical Deployment Considerations 29
5.1.4 Limitations and Future Directions 29

Chapter 6 CONCLUSIONS **31**
6.1 Conclusion and Future Work 31

References **32**

APPENDIX **35**
Appendix A Diagnosis Codes 36

LIST OF TABLES

Table 4.1 Class Distribution 24
Table 4.2 Settings for Different Heart Rates 25
Table 4.3 Performance Metrics of Model Variants Across Different ECG Cycles 27

Table A.1 Diagnosis Codes by Seriousness 36

LIST OF FIGURES

Figure 1.1 Visualizations of different ECG cycles. 6

Figure 2.1 Workflow of calculating the ECG templates; Starting with denoising the raw ECG signals, then finding the R peaks and calculating the heart rate variability. Lastly, by segmenting the R-R intervals, we calculate the average ECG templates and their outliers. 9

Figure 2.2 Overview of our model architecture, detailed in two parts. Part (a) outlines the structure, incorporating a self-attention block, a CNN block, a fully connected block, and an LSTM layer. Part (b) presents our settings, showing the model's complexity. 11

Figure 2.3 The proposed model architecture illustrating the integration of Residual Blocks with SE layers, the Global Attention mechanism, and the Output Layer consisting of three fully connected layers for anomaly detection. . 12

Figure 3.1 Workflow of the real-time ECG processing system, including data acquisition, shifted window segmentation, R-peak detection, heart rate calculation, adaptive model selection based on heart rate (HR), and user alerting for abnormal detection. 16

Figure 3.2 Relation between the relative deadline and the heart rate of ECG segment E_t 17

Figure 4.1 Visualization of Accuracy and Inference Trends, and Confusion Matrix for Advanced Model with Two Cycles. 26

CHAPTER

1

INTRODUCTION

1.1 Definitions

In this dissertation, the following key terms are defined to facilitate understanding of the proposed methodologies:

- **Adaptive Model Selection (AMS):** A dynamic mechanism that selects among multiple pre-trained classification models of varying complexity based on real-time criteria—such as the current heart rate and available computational resources—to ensure that processing deadlines are met without sacrificing diagnostic accuracy.
- **Anytime Model:** An architectural design in which a single neural network is structured with multiple exit points (or intermediate outputs), allowing for valid predictions at various stages of computation. This enables the system to produce quick, preliminary results when processing time is limited, while still permitting more refined analysis if additional computation time is available.
- **Heart Rate Variability (HRV):** The variation in the time intervals between consecutive heartbeats. HRV is a critical parameter in our system as it influences both the segmentation of the ECG signal and the determination of processing deadlines.

- **Cycle:** In the context of this work, a cycle refers to the segment of an ECG signal corresponding to one complete heartbeat—from one R-peak to the next. Fixed cycle segmentation is used to standardize the input size for the anomaly detection models.
- **Global Attention Mechanism:** A method integrated within the model architecture that computes attention weights to dynamically fuse different feature representations (such as ECG features and temporal context), thereby enabling the model to focus on the most diagnostically relevant parts of the input signal.

1.2 Motivations

Cardiovascular diseases (CVDs), responsible for approximately 17.9 million deaths annually, remain the leading cause of mortality worldwide (Organization 2021). This alarming statistic underscores the urgent need for advanced health monitoring solutions that provide continuous and real-time insights into cardiac activity. Early detection and continuous monitoring of heart conditions are critical for reducing mortality rates and improving patient outcomes (Benjamin et al. 2019). In response to this demand, wearable electrocardiogram (ECG) devices have emerged as a promising Cyber-Physical System with Human-in-the-loop technology, offering the potential to detect and interact with heart abnormalities promptly and noninvasively (Patel et al. 2012). In a typical scenario, the user wears the ECG device to continuously monitor heart health. When an anomaly is detected, the system promptly alerts the user and provides recommended actions. After the user takes these actions, the system re-evaluates the heart status and continues the monitoring cycle, creating an iterative feedback loop for ongoing cardiac health management.

As of 2020, over 722 million medical wearable devices were in use globally, each generating vast amounts of physiological data that require immediate processing (Research 2019). With such high adoption rates, issues like missed detection of critical cardiac events and delays in anomaly identification are inevitable. Moreover, these wearable systems face challenges such as signal noise from motion artifacts, variability in heart rates, and limited computational resources inherent in low-power embedded devices (Clifford et al. 2006). These challenges can degrade the reliability of the devices, erode user trust, and potentially lead to adverse health outcomes.

To address these critical challenges, wearable ECG systems must optimize their processing algorithms to handle the dynamic nature of cardiac signals efficiently. Traditional processing approaches, which often rely on fixed-cycle analysis, are ill-suited to accommodate heart rate variability (HRV) and may result in either unnecessary computational overhead or insufficient

analysis time (Pan and Tompkins 1985; Mohamed Suhail and Abdul Razak 2022). The need for adaptive scheduling arises when dealing with varying R-R intervals—the time between consecutive heartbeats—which can fluctuate significantly due to factors like physical activity or stress (Kleiger and Stein 2005). One-size-fits-all models cannot efficiently process ECG data under these conditions, leading to either compromised performance or resource under-utilization.

We frame this ECG-based CVDs detection problem as a dynamic scheduling and processing problem for real-time ECG analysis on wearable devices. Solving this problem necessitates an adaptive system capable of adjusting processing parameters on the fly, balancing computational load and analytical accuracy while adhering to strict real-time constraints (Buttazzo 2011). This involves designing algorithms to optimize model complexity based on current heart rate and accurately classify cardiac anomalies while operating within resource-constrained systems.

In light of these challenges, we propose a novel model architecture tailored for CVD detection, coupled with an adaptive model selection mechanism (Guo and Baruah 2015) that allows dynamic adjustment of model complexity based on the current heart rate, allocating less execution budget for processing higher heart rate to balance the computational load. By evaluating the model using different numbers of heartbeats as input, we optimize accuracy while maintaining computational efficiency. This adaptive approach leverages advanced signal processing and machine learning techniques to ensure that wearable ECG devices can provide accurate and timely cardiac monitoring without relying on high-performance hardware.

Contributions. Our approach addresses these challenges by developing a novel real-time CVD detection architecture, evaluating its performance with varying numbers of heartbeats to optimize diagnostic accuracy, and implementing an AMS model that dynamically adjusts model complexity to ensure real-time processing efficiency. Our main contributions are summarized as follows:

1. **Proposed Novel Model Architecture:** We design a novel CNN architecture for CVD detection, integrating Residual Blocks and a Global Attention mechanism with SE layers to enhance diagnostic performance.

2. **Performance Optimization:** We evaluate the model's performance using different numbers of heartbeats as input with different model complexities, optimizing the balance between diagnostic accuracy and computational efficiency.

3. **AMS-Anytime Model Integration:** We model the minimum processing unit of the ECG signal segment as an AMS task to capture the HRV. The AMS model dynamically selects

the CVD detection model in real-time based on the current heart rate, ensuring efficient operation under varying physiological conditions.

4. **Comprehensive Evaluation on Embedded Systems:** We conduct extensive experiments on Raspberry Pi 4 using real-world ECG datasets, the PhysioNet 2021 Challenge Database (Reyna et al. 2021), ranking the seriousness of the diseases from a scale of 0 to 4, demonstrating the high diagnostic accuracy while adhering to real-time processing constraints.

With the integration of adaptive technologies and data-driven insights, wearable ECG systems can deliver reliable, efficient health assessments. By optimizing model selection and addressing the dynamic nature of cardiac signals, our system enhances real-time ECG monitoring, improving continuous cardiac care and user confidence in wearable health technologies.

1.3 Background

The advancement of wearable technology has significantly influenced real-time ECG monitoring and heart disease detection. This section reviews existing literature on single-lead ECG processing, real-time anomaly detection, adaptive windowing techniques, dynamic model scaling, and the integration of deep learning architectures for enhanced diagnostic performance.

Single-Lead ECG Processing in Wearable Devices: Single-lead ECG devices have gained prominence due to their compactness and ease of integration into wearable systems. Studies have demonstrated that single-lead ECG can reliably detect various cardiac abnormalities, including arrhythmias (Galli et al. 2016). The simplicity of single-lead systems makes them ideal for continuous monitoring in consumer-grade wearables (Ribeiro et al. 2020). However, the potential decrease in diagnostic accuracy necessitates advanced signal processing and machine learning techniques to compensate for the limited data (Liu et al. 2021).

Real-Time ECG Anomaly Detection: Real-time anomaly detection in ECG signals is critical for timely intervention. Traditional methods rely on threshold-based algorithms, which may not generalize well across diverse populations and noisy environments (Elgendi 2013). Recent advances leverage machine learning, particularly deep learning models, to enhance accuracy and robustness (Rajpurkar et al. 2017). Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been employed to automatically extract complex features from raw ECG data, enabling precise identification of irregular heartbeats (Attia et al. 2019).

Dynamic Model Scaling for Resource-Constrained Devices: Dynamic model scaling adjusts the complexity of machine learning models based on processing requirements and available computational resources, beneficial for wearable devices under power and performance

constraints (Cai et al. 2019). Techniques like model pruning and quantization enable real-time performance without compromising accuracy (Han et al. 2016). By dynamically selecting architectures that balance speed and accuracy, wearable ECG systems maintain continuous monitoring capabilities under varying conditions.

Deep Learning Architectures for ECG Anomaly Detection: Deep learning architectures have revolutionized the detection of abnormal ECGs by providing end-to-end solutions that learn hierarchical characteristics from raw signals. CNNs capture local temporal patterns, while RNNs and Long Short-Term Memory (LSTM) networks model sequential dependencies (Hannun et al. 2019). Hybrid architectures combining CNNs with attention mechanisms enhance feature extraction and focus on critical ECG segments (Islam et al. 2024). These advanced models achieve high accuracy in classifying cardiac conditions such as atrial fibrillation and ventricular tachycardia (Strodthoff et al. 2021).

Integration of Adaptive Techniques in ECG Monitoring Systems: The integration of adaptive techniques such as shifted window approaches and dynamic model scaling within ECG monitoring systems represents a novel advance. Althoughugh studies have explored these techniques individually, their combined application freal-timeme single-lead ECG analysis remainunderexploredd (Gupta et al. 2024). Our proposed system builds upon these foundations by implementing an adaptive shifted window mechanism coupled with dynamic model selection based on heart rate variability. This integration ensures efficient resource utilization and maintains high diagnostic performance, addressing critical challenges in real-time ECG monitoring on wearable devices.

1.4 Problem Statement

Wearable ECG devices are designed to provide continuous cardiac monitoring by capturing the heart's electrical signals in real-time. However, these devices encounter significant challenges due to heart rate variability (HRV) and the necessity to operate within strict real-time processing constraints, especially given the limited computational resources of platforms like the Raspberry Pi 4.

Heart Rate Variability and Cycle Number: HRV refers to the fluctuations in the time intervals between consecutive heartbeats (R-R intervals) (Kleiger and Stein 2005). It is influenced by factors such as physical activity, stress, and individual health conditions. This variability affects the duration of cardiac cycles, complicating the use of a fixed number of cycles for ECG signal analysis.

In this study, we segment the ECG signal into a fixed number of cycles for processing. As illustrated in Figure 1.1, a "cycle" corresponds to a single heartbeat, typically defined from one

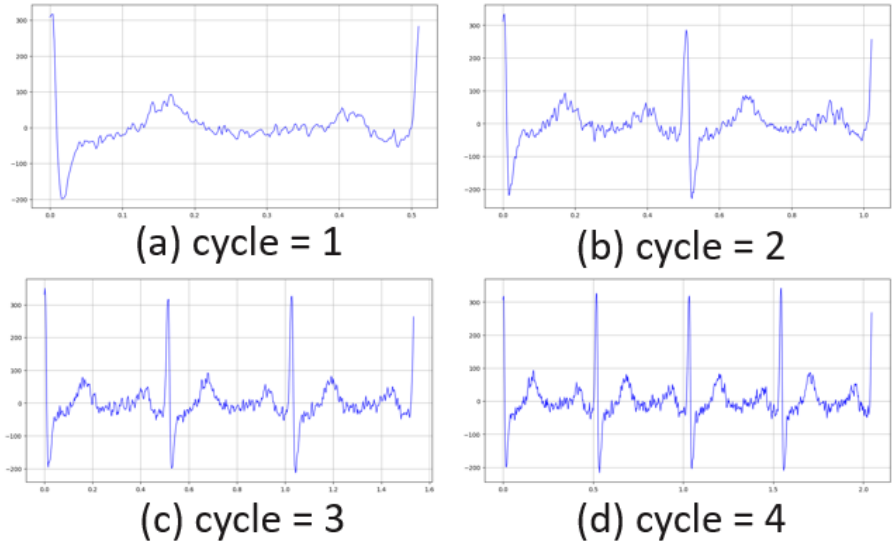


Figure 1.1: Visualizations of different ECG cycles.

R peak to the next in the ECG waveform. Selecting an appropriate number of cycles is crucial because it must capture sufficient cardiac information for accurate anomaly detection while minimizing the introduction of noise and artifacts. A higher number of cycles may encompass more comprehensive cardiac activity but could also include more noise and irrelevant variations, potentially hindering anomaly detection performance. Conversely, a lower number of cycles reduces computational overhead and noise but may risk missing subtle diagnostic features. Therefore, determining the optimal cycle number is essential for balancing information capture and processing efficiency.

Adaptive Model Selection: Implementing algorithms for disease anomaly detection on resource-constrained wearable devices poses significant challenges (Buttazzo 2011). To address this, we employ an adaptive model selection approach, where the complexity of the anomaly detection model M_t adjusts based on the current heart rate.

At higher heart rates, the system requires faster processing to meet real-time constraints, necessitating the use of simpler and less compute-required models. Conversely, at lower heart rates, more complex models can be utilized to enhance anomaly detection accuracy without violating timing deadlines. This adaptive selection ensures that the system remains both responsive and accurate, adapting to the user's physiological state in real-time.

1.4.1 Problem Formulation

Our objective is to design an adaptive CVD anomaly detection system that maximizes diagnostic accuracy while adhering to real-time processing deadlines inherent in wearable devices like

the Raspberry Pi 4. To achieve this, we formulate an optimization problem that adjusts the anomaly detection model to find the optimal balance between accuracy and efficiency.
Decision Variable: At each processing iteration t, we determine the optimal model $M_t \in \{M_{\text{Adv}}, M_{\text{Mod}}, M_{\text{LW}}\}$, where $M_{\text{Adv}}, M_{\text{Mod}}$,
M_{LW} are an advanced, moderate, and lightweight learning model respectively.
Optimization Problem: We seek to select M_t that maximizes the diagnostic accuracy A while ensuring that the processing time $T_{\text{proc}}(M_t)$ does not exceed the deadline $D(\text{HR})$ determined by the current heart rate:

$$\begin{aligned} &\arg\max_{M_t} \quad A = \text{Accuracy}(M_t) \\ &\text{s.t.} \quad T_{\text{proc}}(M_t) \leq D(\text{HR}); \quad M_t \in \{M_{\text{Adv}}, M_{\text{Mod}}, M_{\text{LW}}\} \end{aligned} \tag{1.1}$$

Explanation:

- The **objective function** $A = \text{Accuracy}(M_t)$ represents the diagnostic accuracy achieved by using model M_t.
- The **processing time constraint** $T_{\text{proc}}(M_t) \leq D(\text{HR})$ ensures real-time operation by keeping the processing time within the allowable limit $D(\text{HR})$, which is calculated based on the current heart rate and HRV.
- The **model selection** restricts M_t to one of the predefined models—Advanced, Moderate, or Lightweight—suitable for implementation on the Raspberry Pi 4.

By solving this optimization problem at each iteration, the system adaptively selects the most appropriate anomaly detection model in response to changing heart rates, ensuring high diagnostic accuracy without violating real-time constraints.

1.4.2 Objective

The core problem is to develop an adaptive CVD anomaly detection system that selects the appropriate anomaly detection model to maximize diagnostic accuracy while satisfying real-time processing constraints. The proposed framework leverages an adaptive model selection mechanism with anytime model to dynamically adjust model complexity based on the current heart rate, ensuring efficient and accurate real-time ECG monitoring on wearable devices.

CHAPTER

2

MODEL DESIGN

2.1 Approach 1: Self-Attention CNN-LSTM

2.1.1 ECG Pattern Recognition

In this section, we introduce the approaches for extracting the ECG templates and their outliers to mitigate the artifacts and improve the robustness of heart disease detection.

ECG Templates Detection

Shown in Figure 2.1, the extraction of individual heartbeats is centered around the identification of R-peaks within the ECG signal. For each R-peak located at position R_i, a segment S_i of the ECG signal is defined as follows:

$$S_i = \mathrm{ECG}\left[R_i - \frac{T}{2} : R_i + \frac{T}{2}\right] \tag{2.1}$$

where T is the duration of the window around each R-peak, designed to encompass the entire heartbeat. The mean ECG template $\bar{A}$ is then computed by averaging these segments across all detected heartbeats:

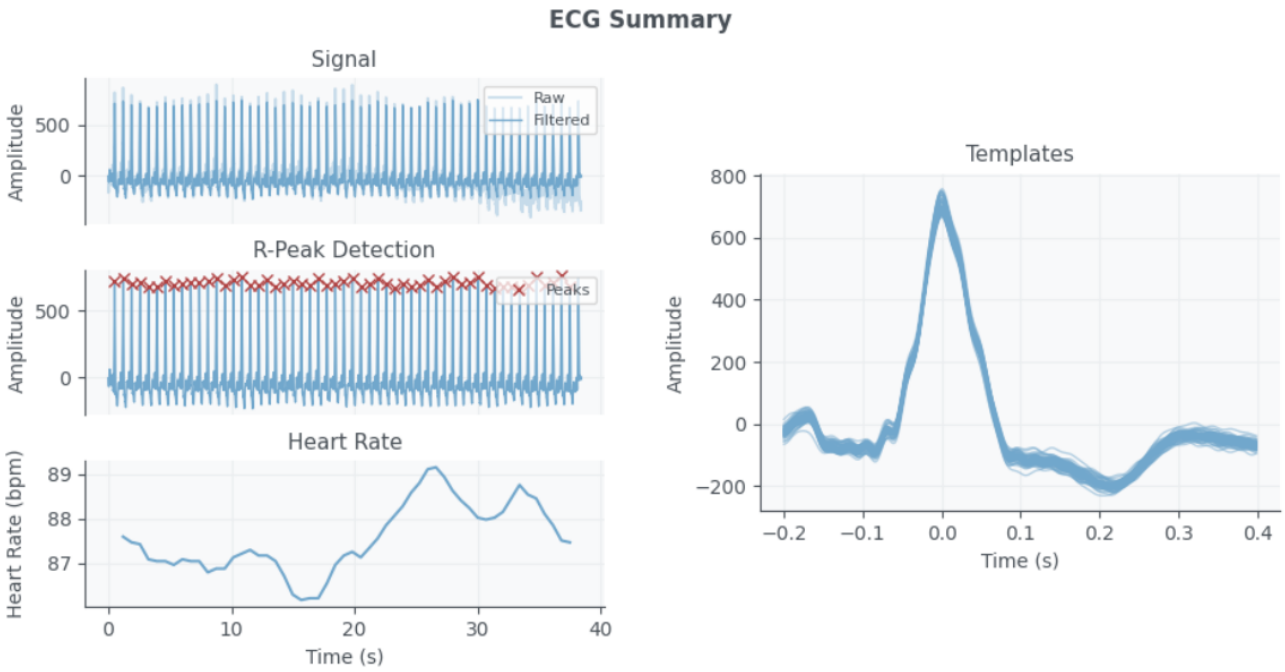


Figure 2.1: Workflow of calculating the ECG templates; Starting with denoising the raw ECG signals, then finding the R peaks and calculating the heart rate variability. Lastly, by segmenting the R-R intervals, we calculate the average ECG templates and their outliers.

$$\bar{A} = \frac{1}{N} \sum_{i=1}^{N} S_i \tag{2.2}$$

This process not only reduces noise, but also enhances the clarity of the ECG signal, providing a reliable basis for further analysis.

Outlier Detection in ECG Pattern Recognition.

Our approach to outlier detection within ECG pattern recognition starts with the computation of deviations between each resampled ECG template and the mean template. These deviations are quantified by calculating the Euclidean distance for each template from the mean, encapsulated by the following equation:

$$\text{deviation} = \sqrt{\sum (\text{resampled\_templates} - \text{mean\_templates})^2} \tag{2.3}$$

Based on these deviations, we establish a threshold for detecting outliers, defined as the mean of the deviations plus 2.1 times their standard deviation:

$$\text{threshold} = \text{mean}(\text{deviations}) + 2.1 \times \text{std}(\text{deviations}) \tag{2.4}$$

When the deviations exceed this threshold, the corresponding indices are marked as outliers. If outliers are identified (i.e., the outlier indices array is not empty), we extract the outlier templates from the ECG dataset using these indices. The average of these outlier templates is calculated to form a representative outlier pattern, which is then resampled to a predetermined number of input data points, ensuring consistency across the analysis.

Kurtosis Signal Quality Index (kSQI).

To assess the quality of the ECG signal, we employ the Kurtosis Signal Quality Index (kSQI). Kurtosis (K) is calculated using the formula:

$$K = \frac{N \sum_{i=1}^{N} (x_i - \bar{x})^4}{(\sum_{i=1}^{N} (x_i - \bar{x})^2)^2} - 3 \tag{2.5}$$

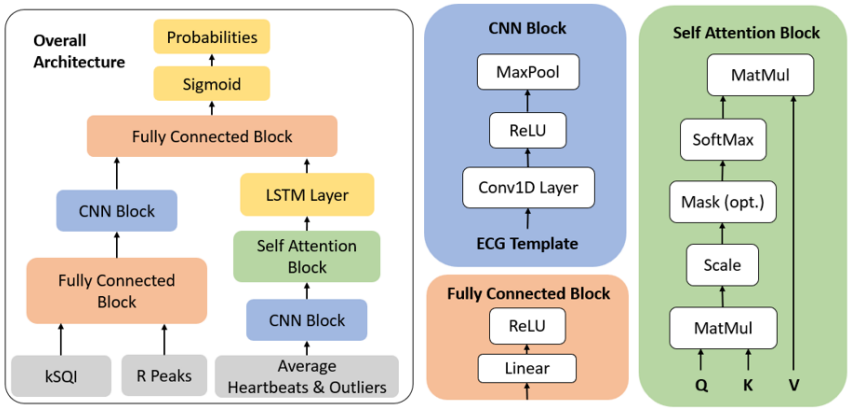
where x_i represents the amplitude values of the ECG signal, $\bar{x}$ is the mean amplitude, and N is the total number of samples. A high kurtosis value indicates potential noise or outliers, suggesting a low-quality ECG signal, while a value near 3 implies a high-quality, less noisy signal.

2.1.2 Model Design

Figure 2.2 illustrates the proposed model, which includes an attention mechanism for ECG data analysis. This model integrates CNNs for spatial feature extraction and LSTMs for handling time-series data, effectively capturing ECG signal dynamics. The attention mechanism further allows the model to focus on relevant ECG segments, improving pattern recognition in cardiac signals.

The model begins by analyzing initial inputs including the kSQI, R-peaks, and their outliers, feeding these into a fully connected layer (Initial FC). This is followed by a 1D convolutional layer (Initial Conv1d) that extracts features from these inputs. Meanwhile, the mean of ECG heartbeats and their outliers are processed through two 1D convolutional layers (Conv1 and Conv2), extracting deeper features, which are then passed to an attention block for refinement.

Subsequent processing involves two layers of LSTMs that integrate the detailed features with initial insights, offering a comprehensive analysis that encompasses both basic signal characteristics and complex temporal relationships. The combined features from the kSQI, R-peaks, their outliers, and the LSTM-processed data are then fed into a fully connected layer.



(a) Model architecture

Hyperparameter	Value
Initial FC Output Size	32
Initial Conv1d Output	32
FC After Conv Output Size	64
LSTM Input Size	16
LSTM Hidden Size	256
LSTM Number of Layers	2
LSTM Bidirectional	False
Conv1 Output Channels	8
Conv2 Output Channels	16
Attention Channels	16
FC1 Input Size	320
FC1 Output Size	256
Probability Output Size	129

(b) Model hyperparameters

Figure 2.2: Overview of our model architecture, detailed in two parts. Part (a) outlines the structure, incorporating a self-attention block, a CNN block, a fully connected block, and an LSTM layer. Part (b) presents our settings, showing the model's complexity.

A sigmoid function is applied at the final stage for multi-label classification, enabling the identification of multiple cardiac conditions simultaneously in a probabilistic manner. This structured approach ensures a thorough analysis of ECG data.

2.2 Approach 2: Attention-based SE-ResNet

The proposed disease detection model, shown in Figure 2.3, introduces significant advancements tailored for supervised CVD disease detection and real-time processing efficiency. The model leverages convolutional neural network (CNN) components enhanced with Residual Blocks, Squeeze-and-Excitation (SE) layers, and a Global Attention mechanism to improve both accuracy and computational efficiency in real-time CVD detection.

2.2.1 Model Workflow

In our proposed model architecture, α denotes the target sequence length ($\alpha = \text{target\_length}$), β represents the number of ECG cycles per sample ($\beta = \text{cycle\_num}$), and γ signifies the number of target classes ($\gamma = \text{num\_classes}$).

Our model processes inputs consisting of ECG cycles, each containing $256 \times \beta$ data points, along with corresponding period information comprising β data points. The architectural flow is illustrated in Figure 2.3, highlighting the integration of key components that collectively enhance feature extraction, adaptive focusing, and anomaly detection performance.

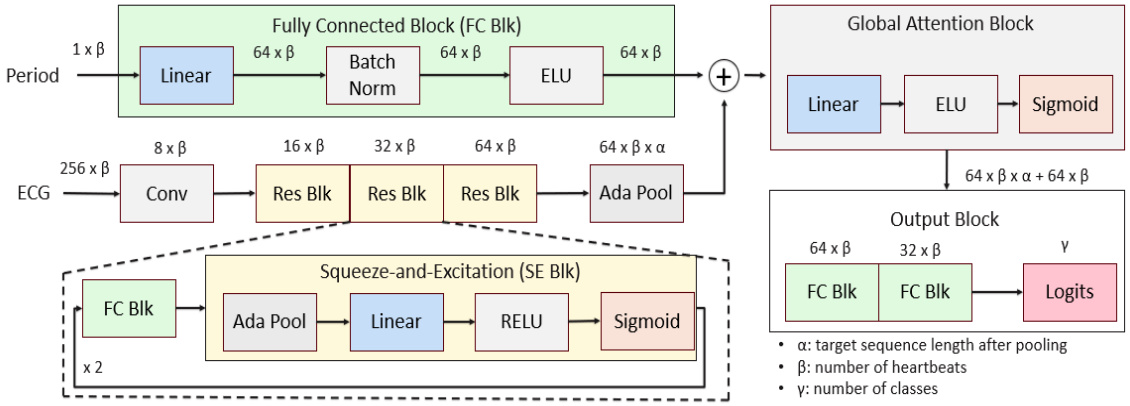


Figure 2.3: The proposed model architecture illustrating the integration of Residual Blocks with SE layers, the Global Attention mechanism, and the Output Layer consisting of three fully connected layers for anomaly detection.

The disease detection pipeline begins with ECG data undergoing feature extraction via a convolutional layer, followed by Residual Blocks enhanced with SE layers to prioritize important features. Adaptive pooling standardizes the feature maps across varying input lengths, while period features are projected into a higher-dimensional space for seamless integration with the ECG data. Next, a Global Attention mechanism fuses ECG and period features, allowing the model to focus on critical signal regions based on temporal and contextual information. The combined features are passed through fully connected layers, producing classification logits for γ disease classes. This design balances computational efficiency with dynamic feature emphasis, ensuring high diagnostic accuracy in real-time applications.

2.2.2 Residual Blocks with Squeeze-and-Excitation Layers

A fundamental advancement in our model is the incorporation of SE layers within Residual Blocks, which enhances feature recalibration. This allows the network to emphasize informative features while suppressing less relevant ones. Mathematically, each Residual Block with an SE layer can be described as:

$$\mathbf{y} = \mathscr{F}(\mathbf{x}, \{\mathbf{W}_i\}) \otimes \mathbf{s} + \mathbf{x}, \tag{2.6}$$

In our formulation, $\mathbf{x}$ represents the input feature map, $\mathscr{F}(\mathbf{x}, \{\mathbf{W}_i\})$ denotes the residual function composed of convolutional operations, $\mathbf{W}_i$ are the learnable parameters, and $\mathbf{s} \in \mathbb{R}^C$ is the scaling vector obtained through the SE layer.

The scaling vector $\mathbf{s}$ is computed as:

$$\mathbf{s} = \sigma\left(\mathbf{W}_2 \cdot \delta\left(\mathbf{W}_1 \cdot \text{AvgPool}(\mathbf{x})\right)\right), \tag{2.7}$$

$$\mathbf{s} \in \mathbb{R}^C, \tag{2.8}$$

AvgPool($\mathbf{x}$) performs global average pooling on the input feature map, δ denotes the ReLU activation function, σ represents the sigmoid activation function, and $\mathbf{W}_1$ and $\mathbf{W}_2$ are the weight

The element-wise multiplication $\otimes$ applies the recalibrated weights to the residual output, enhancing the network's ability to focus on salient features while mitigating the vanishing gradient problem inherent in deep architectures.

2.2.3 Global Attention Mechanism

Building upon the feature extraction capabilities, the Global Attention mechanism is a pivotal advancement that effectively integrates ECG features with period features to enhance the model's discriminative capability. This mechanism operates by computing attention weights that dynamically emphasize critical regions of the ECG signal based on both intrinsic ECG characteristics and temporal context provided by period features.

Formally, let $\mathbf{f}_{\text{ECG}} \in \mathbb{R}^{d_{\text{ECG}}}$ denote the flattened ECG feature vector and $\mathbf{f}_{\text{period}} \in \mathbb{R}^{d_{\text{period}}}$ represent the period feature vector. The combined feature vector is defined as:

$$\mathbf{f}_{\text{combined}} = \begin{bmatrix} \mathbf{f}_{\text{ECG}} \\ \mathbf{f}_{\text{period}} \end{bmatrix} \in \mathbb{R}^{d_{\text{ECG}}+d_{\text{period}}}. \tag{2.9}$$

The attention weights $\mathbf{a} \in \mathbb{R}^{d_{\text{ECG}}}$ are computed through a two-layer fully connected network:

$$\mathbf{h} = \text{ELU}(\mathbf{W}_1\mathbf{f}_{\text{combined}} + \mathbf{b}_1), \tag{2.10}$$

$$\mathbf{a} = \sigma(\mathbf{W}_2\mathbf{h} + \mathbf{b}_2), \tag{2.11}$$

where: $\mathbf{W}_1 \in \mathbb{R}^{m\times(d_{\text{ECG}}+d_{\text{period}})}$ and $\mathbf{W}_2 \in \mathbb{R}^{d_{\text{ECG}}\times m}$ are the weight matrices, where m represents the dimensionality of the intermediate layer, while $\mathbf{b}_1$ and $\mathbf{b}_2$ denote the bias vectors, and σ is the sigmoid activation function.

The computed attention weights are then applied to the ECG feature map $\mathbf{F}_{\text{ECG}} \in \mathbb{R}^{d_{\text{ECG}}\times T}$ (where T represents the temporal dimension):

$$\mathbf{F}_{\text{attended}} = \mathbf{a} \otimes \mathbf{F}_{\text{ECG}}, \tag{2.12}$$

where $\otimes$ denotes element-wise multiplication. This operation effectively highlights regions of the ECG signal that are most indicative of potential anomalies, guided by both the ECG data and the period context.

2.2.4 Output Layer

Following the Global Attention mechanism, the attended feature map undergoes further processing through three fully connected layers to generate the final anomaly detection logits. The first fully connected layer reduces the dimensionality of the attended features, facilitating efficient computation, while the subsequent layers progressively map these features to the desired number of classes, denoted by γ.

Mathematically, the output layers can be described as:

$$\mathbf{h}_1 = \text{ReLU}(\mathbf{W}_3\mathbf{F}_{\text{attended}} + \mathbf{b}_3), \tag{2.13}$$

$$\mathbf{h}_2 = \text{ReLU}(\mathbf{W}_4\mathbf{h}_1 + \mathbf{b}_4), \tag{2.14}$$

$$\text{Logits} = \mathbf{W}_5\mathbf{h}_2 + \mathbf{b}_5, \tag{2.15}$$

where: $\mathbf{W}_3 \in \mathbb{R}^{64\beta\times(64\beta\times\alpha)}$, $\mathbf{W}_4 \in \mathbb{R}^{32\beta\times64\beta}$, $\mathbf{W}_5 \in \mathbb{R}^{\gamma\times32\beta}$. In addition, $\mathbf{b}_3$, $\mathbf{b}_4$, and $\mathbf{b}_5$ are the corresponding bias vectors.

The ReLU activation function introduces non-linearity in the hidden layers, enhancing the model's capacity to learn complex patterns. The final logits are then used to compute class probabilities through a softmax function perioding the training and inference phases. This three-layer fully connected output structure ensures that the model can effectively translate the attended feature representations into precise disease anomaly detections, accommodating a wide range of cardiac conditions as defined by the γ classes.

CHAPTER

3

FRAMEWORK

3.1 System Design

The proposed system, shown in Figure 3.1, is designed to process single-lead ECG data in real time, ensuring timely and accurate heart rate monitoring and disease detection under varying physiological conditions. The workflow comprises several key stages, as detailed below:

3.1.1 Shifted Window Approach

ECG data is continuously captured from the user using a single-lead configuration. To manage the data efficiently and facilitate real-time processing, a shifted window approach is employed. This method segments the incoming ECG signal into overlapping time windows, thereby limiting the temporal scope for subsequent analysis and enabling continuous monitoring without significant latency.

Within each time window, the system performs R-peak detection, a critical step for identifying individual heartbeats. Accurate detection of R-peaks allows for the extraction of heartbeat cycles from the ECG signal. Various algorithms can be utilized to ensure reliable identification of these characteristic points in the ECG waveform. In our study, we use a well-recognized R-peak detection algorithm, Hamilton's algorithm (Hamilton 2002), to conduct experiments.

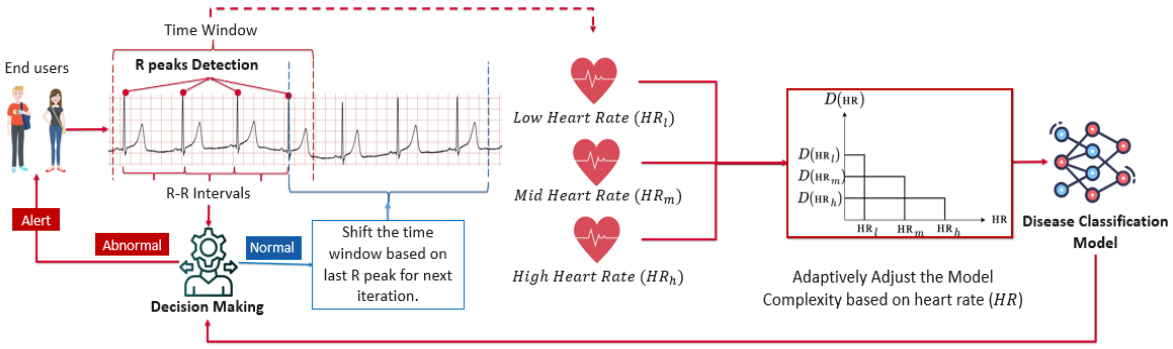


Figure 3.1: Workflow of the real-time ECG processing system, including data acquisition, shifted window segmentation, R-peak detection, heart rate calculation, adaptive model selection based on heart rate (HR), and user alerting for abnormal detection.

Following R-peak detection, the system calculates the heart rate by determining the number of R-peaks within the defined time window. The heart rate (HR) is expressed in beats per minute (BPM) and serves as a key parameter for subsequent model selection. The calculation can be formalized as:

$$\mathrm{HR} = \frac{N_{\text{R-peaks}}}{T} \times 60\,\mathrm{BPM}, \tag{3.1}$$

where N_R-peaks is the number of detected R-peaks and T is the length of the time window in seconds.

Fixed Cycle Segmentation: Let $E(t_1, t_2)$ denote the ECG signal segment from time t_1 to t_2. The signal is segmented into a fixed number of cycles C for processing. Each segment E_t is computed as:

$$E_t = E(t - C \cdot L,\ t), \tag{3.2}$$

where L represents the length of each cycle segment. By maintaining a fixed cycle number, the framework ensures a consistent data input size for the anomaly detection models, simplifying the processing pipeline and enhancing computational efficiency. This fixed segmentation strategy reduces variability in the input data, allowing the models to focus on capturing the most salient features associated with cardiac conditions without being confounded by fluctuating cycle lengths.

Task Model of E_t: The ECG segment at t^{th} instant E_t is highly dependent on the user's current heart rate, i.e., segment length $C \cdot L$ directly related to the heart rate or ECG cycle. Depending on the HRV, the subsequent sampling period of ECG segments can vary significantly, which results in variable processing window length to process E_t for detecting CVDs. To maintain real-time processing constraint, each E_t needs to assign a relative deadline $D_t(\mathrm{HR})$ that is related to the user's current heart rate at the sampling instant of E_t. Fig. 3.2 illustrates the relation between the

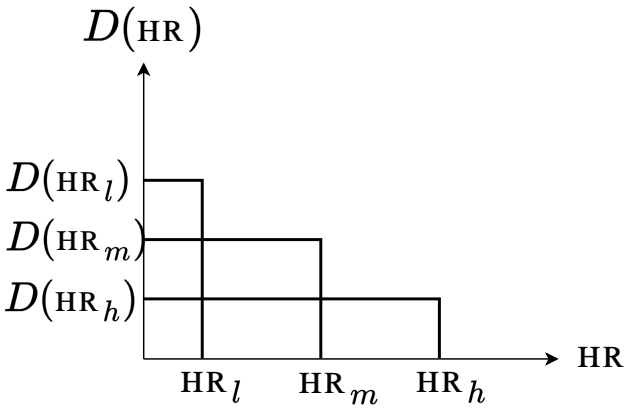


Figure 3.2: Relation between the relative deadline and the heart rate of ECG segment E_t

deadline and heart rate of ECG segment E_t. To process E_t within $D_t(\mathrm{HR})$, there is a worst-case execution budget bound which should be respected while selecting an ML model for E_t. Let us model E_t as three-tuple $(C_t(\mathrm{HR}), D_t(\mathrm{HR}), \mathrm{HR})$, where $D_t(\mathrm{HR})$ is the relative deadline, $C_t(\mathrm{HR})$ is worst-case execution time and HR is the heart rate (computed online) corresponding to the ECG segment E_t. Note that the next sampling instant of the ECG signal is at or after $t + D_t(\mathrm{HR})$.

3.1.2 Adaptive Model Selection

Real-time ECG monitoring, especially on edge devices, must accommodate diverse operating conditions that affect both performance requirements and computational budgets.

To accommodate the diverse range of heart rates and ensure real-time performance, the system adopts an adaptive model selection strategy based on the calculated BPM:

- **(1) High Heart Rate** : A lightweight model with rapid inference speed is employed to process the ECG cycles, facilitating immediate feedback and minimizing processing delays.
- **(2) Normal Heart Rate** : A moderately sized model is utilized to balance inference speed and diagnostic accuracy, providing reliable detection without compromising real-time constraints.
- **(3) Low Heart Rate** : A more comprehensive and larger model is activated to perform detailed disease detection, accepting longer processing times due to the increased complexity and the necessity for thorough analysis under critical conditions.

This tiered approach ensures that the system maintains optimal performance across varying heart rate conditions by dynamically allocating computational resources based on the

immediate needs of the user. Note that the total execution time for processing E_t depends on the ML model, and hence, a model should be selected such that the total execution time is $\leq C_t(\text{HR})$.

We propose two complementary strategies for achieving high diagnostic accuracy under such conditions: (1) an AMS mechanism that selects among multiple pre-trained models based on heart rate or other resource constraints, and (2) an Anytime Model with Parameter Sharing, which consolidates all model variants into a single scalable network to reduce memory overhead on edge devices.

Adaptive Model Selection provides a runtime mechanism that intelligently chooses between multiple classification models of varying complexity. For instance, in high-heart-rate scenarios where the segment length or sampling rate may be larger—and processor utilization is higher—a lightweight model is preferred to ensure deadlines are met. Conversely, at normal or low heart rates, the system can afford a more advanced model to maximize accuracy. Specifically:

1. **Model Pool:** We pre-train three variants (e.g., LIGHTWEIGHT, MODERATE, and ADVANCED) with progressively higher parameter counts and diagnostic capabilities.

2. **Selection Criterion:** During online execution, the measured heart rate or resource availability triggers a choice:

    - If HR exceeds a high threshold or CPU load is high, run the lightweight model quickly.
    - If HR falls below a lower threshold, use the advanced model for thorough analysis.
    - Otherwise, choose the moderate model for a balanced trade-off.

3. **Real-Time Guarantees:** By selecting the appropriate model at runtime, AMS enforces that each ECG segment is processed before the next segment arrives, effectively accommodating changing heart rates and resource conditions.

This approach simplifies scheduling: at each sampling instant, we pick the model whose worst-case execution time satisfies the current task deadline, ensuring continuous ECG analysis without missing beats or incurring excessive computational delay.

3.1.3 Anytime Model with Parameter Sharing

To meet the real-time and memory constraints of embedded systems, we propose a unified anytime model that dynamically adjusts its inference depth based on the measured heart rate.

Instead of deploying three separate models (lightweight, moderate, and advanced), this single network shares parameters across all processing stages and offers multiple early exits. The design enables the system to produce a fast, approximate prediction when the heart rate is high (and processing time is limited), while allowing deeper inference for enhanced accuracy when the heart rate is lower.

Let the input ECG signal be denoted by $\mathbf{x}$. The network processes $\mathbf{x}$ through a series of K sequential blocks $L_1, L_2, \ldots, L_K$, yielding intermediate feature representations:

$$\mathbf{h}_0 = \mathbf{x}, \quad \mathbf{h}_i = L_i(\mathbf{h}_{i-1}), \quad i = 1, 2, \ldots, K.$$

At each block i, an early-exit classifier g_i produces a prediction:

$$\mathbf{y}_i = g_i(\mathbf{h}_i).$$

We define two heart rate thresholds, $\mathrm{HR}_{\mathrm{high}}$ and $\mathrm{HR}_{\mathrm{low}}$ (with $\mathrm{HR}_{\mathrm{high}} > \mathrm{HR}_{\mathrm{low}}$). When the measured heart rate is high ($\mathrm{HR} \geq \mathrm{HR}_{\mathrm{high}}$), fast processing is critical, so the system exits early—using an intermediate prediction $\mathbf{y}_i$ for some $i < K$—to meet real-time constraints. In the case where the heart rate falls in the intermediate range ($\mathrm{HR}_{\mathrm{low}} \leq \mathrm{HR} < \mathrm{HR}_{\mathrm{high}}$), the system opts for an intermediate exit that achieves a balance between speed and accuracy. Conversely, when the heart rate is low ($\mathrm{HR} < \mathrm{HR}_{\mathrm{low}}$), and more processing time is available, the system processes all K blocks and outputs the final prediction $\mathbf{y}_K$ to maximize diagnostic accuracy.

This design is motivated by several practical benefits. First, the network's progressive structure allows it to provide valid intermediate outputs: early exits enable rapid inference when necessary, while deeper exits improve accuracy when time permits. Second, sharing the backbone layers among all exits reduces the overall memory footprint compared to maintaining multiple independent models, and it simplifies model maintenance since enhancements to the shared layers benefit every exit. Finally, by dynamically adjusting its computational effort based on real-time physiological measurements, the model achieves a flexible balance between speed and accuracy.

During training, each exit is supervised with its own classification loss. Denoting the loss at exit i by $\mathscr{L}_i(\mathbf{y}_i, \mathbf{y}^*)$ (with $\mathbf{y}^*$ as the ground truth), the overall loss is defined as:

$$\mathscr{L} = \sum_{i=1}^{K} \alpha_i \mathscr{L}_i(\mathbf{y}_i, \mathbf{y}^*),$$

where α_i are weighting factors that control the relative importance of each exit.

The inference procedure for this anytime model is outlined in the following pseudocode:

Algorithm 1: Anytime Inference with Adaptive Early Exits

Input : ECG sample $\mathbf{x}$, measured heart rate HR

Output : Classification output $\mathbf{y}$

Initialize: $\mathbf{h}_0 \leftarrow \mathbf{x}$
for $i = 1$ **to** K **do**
    Compute $\mathbf{h}_i \leftarrow L_i(\mathbf{h}_{i-1})$
    if $\text{HR} \geq \text{HR}_{high}$ ***and*** *early exit condition met at block* i **then**
        return $\mathbf{y} \leftarrow g_i(\mathbf{h}_i)$
    else
        if $\text{HR}_{low} \leq \text{HR} < \text{HR}_{high}$ ***and*** *intermediate exit condition met at block* i **then**
            return $\mathbf{y} \leftarrow g_i(\mathbf{h}_i)$
return $\mathbf{y} \leftarrow g_K(\mathbf{h}_K)$

In summary, this anytime model with parameter sharing provides a unified and efficient solution for real-time ECG classification on embedded systems. By dynamically choosing the appropriate exit based on the current heart rate, the model adapts its inference depth to balance the competing demands of speed and diagnostic accuracy.

3.2 Schedulability Analysis for Real-time Processing

In this section, we present our scheduling model, describe the scheduling algorithm, and analyze the schedulability of the overall system workload, which includes both the adaptive ECG processing task and other sporadic tasks. We assume a uniprocessor system for this analysis.

3.2.1 System Model

We consider a recurrent workload Γ that consists of two types of tasks: the adaptive ECG task E_t and a set of sporadic tasks $\{\tau_1, \tau_2, \ldots, \tau_n\}$ representing all other system processes. The ECG task E_t is adaptive in nature, meaning its execution characteristics vary with the user's heart rate. At the t^{th} sampling instance, E_t is characterized by the tuple

$$(C_t(\text{HR}),\ D_t(\text{HR}),\ \text{HR}),$$

where:

- $C_t(\text{HR})$ is the worst-case execution time (WCET) for processing the ECG segment,

- $D_t(\text{HR})$ is the relative deadline by which the processing must be completed, and
- HR is the user's heart rate at that instance.

Both $C_t(\text{HR})$ and $D_t(\text{HR})$ are functions of HR, as illustrated in Fig. 3.2. The ECG task may produce an infinite sequence of instances, with each instance released at least $D_t(\text{HR})$ time units apart.

Each sporadic task τ_i is modeled by the pair (C_i, T_i), where C_i is its WCET and T_i is its period (which also serves as the relative deadline). Like E_t, sporadic tasks may also generate an infinite number of instances, with a minimum inter-arrival time of T_i.

3.2.2 Scheduling Algorithm

All tasks in Γ are scheduled on a uniprocessor using a preemptive Earliest Deadline First (EDF) algorithm. Under EDF, if there is a set of ready tasks R, the scheduler selects the task instance τ_k whose absolute deadline d_k is the smallest:

$$\tau_k = \arg\min_{\tau_i \in R}\{d_i\}.$$

In our system, each instance of the adaptive ECG task E_t has a deadline $D_t(\text{HR})$ that varies with the user's heart rate, while each sporadic task τ_i has a fixed deadline equal to its period T_i. This dynamic scheduling approach is particularly well-suited for handling the variable execution times and release intervals of both the adaptive ECG task and the sporadic tasks.

3.2.3 Schedulability Analysis

The schedulability of the system is determined using the classical Liu and Layland utilization test (Liu and Layland 1973). For each sporadic task τ_i, the utilization is defined as:

$$U(\tau_i) = \frac{C_i}{T_i}.$$

For the adaptive ECG task E_t, the utilization is given by:

$$U(E_t) = \frac{C_t(\text{HR})}{D_t(\text{HR})}.$$

Because E_t's parameters depend on the heart rate, its utilization can vary; however, both $C_t(\text{HR})$ and $D_t(\text{HR})$ are restricted to a finite set of values determined by predefined heart rate thresholds (as shown in Fig. 3.2). Thus, the maximum utilization of E_t over all instances can be computed efficiently as $\max_t U(E_t)$.

The system is considered schedulable under EDF if the total utilization does not exceed 1:

$$\sum_{i=1}^{n} \frac{C_i}{T_i} + \max_t \frac{C_t(\mathrm{HR})}{D_t(\mathrm{HR})} \leq 1.$$

This criterion ensures that, even in the worst-case scenario for the ECG task, the combined workload of all tasks can be scheduled without missing deadlines.

By clearly defining the system model, the scheduling strategy, and the utilization-based schedulability condition, this analysis provides a robust framework for ensuring that our real-time ECG processing system operates reliably on resource-constrained embedded devices.

CHAPTER

4

EXPERIMENTS AND RESULTS

4.1 Experiments and Results

4.1.1 Experimental Implementation

Dataset

We utilized the PhysioNet 2021 Challenge Database, which contains annotated twelve-lead ECG recordings from six sources across four countries and three continents, comprising over 100,000 recordings and covering 133 distinct disease classes Reyna et al. (2021). For our study, we filtered the dataset to include only the 72 disease classes that can be diagnosed using Lead I ECG data. Each disease was ranked according to its clinical severity, with a seriousness score assigned from 0 to 4, where 4 represents the most severe condition; these scores were used as labels. More information is provided in Table A.1.

To ensure beat-level evaluation precision, we refined the dataset by selecting only the recordings with a single disease label, resulting in a dataset of 22,359 single-label ECG recordings ranging from 5 to 1800 seconds in length. By excluding multi-label recordings, we eliminate ambiguity in anomaly detection, enabling more accurate model training and evaluation at the individual beat level.

Shown in Table 4.1, we split the dataset into 80% training and 20% testing, and further split

Table 4.1: Class Distribution

Label	Training Set	Validation Set	Test Set
Label 0	145,866	36,467	45,583
Label 1	107,908	26,977	33,722
Label 2	42,760	10,690	13,363
Label 3	43,882	10,971	13,713
Label 4	10,106	2,526	3,158

the training portion (80%) into 80% for actual training (64% of the original data) and 20% for validation (16% of the original data). This results in a final distribution of 64% for training, 16% for validation, and 20% for testing across all labels.

The overall label distribution percentages are consistent across these splits, maintaining the original class imbalance. Label 0 comprises approximately 33% of the entire dataset, while Label 1 makes up about 24.6%. Labels 2 and 3 account for around 9.8% and 10%, respectively. Label 4, the least represented class, makes up only about 2.3% of the dataset. This imbalance mirrors real-world scenarios, where data for highly severe diseases is limited.

Experimental Setup

We trained our models on a NVIDIA 3090 GPU and tested them on an embedded device, Raspberry Pi 4, to simulate real-world scenarios.

To prepare the ECG recordings for detecting disease anomalies, we improved signal quality and consistency through several preprocessing steps. We first identified R-peaks in the ECG signals using Hamilton's algorithm from the BioSPPY library Carreiras et al. (15), which allowed us to segment the continuous data into individual heartbeats or cardiac cycles. Each cycle was then resampled to have 256 data points to ensure uniform length for consistent model input. We calculated the time intervals between consecutive R-peaks to capture heart rate variability, a critical feature for detecting heart disease anomalies. Finally, the preprocessed ECG cycles and their corresponding heart rate variability features were organized into a format suitable for model training, aligning each input with its respective disease label. These steps ensured we had high-quality, consistent data for effective model training and evaluation.

Experiments

We assess our experiments in two main aspects:
Model Performance Evaluation: We assessed three different models in Approach 2 to understand how various architectural elements affect diagnostic accuracy and computational

efficiency:

- **Advanced Model:** Incorporates Residual Blocks, SE layers, and a Global Attention mechanism.
- **Moderate Model:** Similar to the Advanced Model but without the Global Attention mechanism to evaluate its impact.
- **Lightweight Model:** A simplified version that reduces the number of parameters and excludes both the Global Attention mechanism and SE layers.

We tested each model variant using 1, 2, 3, and 4 cardiac cycles to explore how the amount of temporal context influences diagnostic performance. Additionally, we compared our models with two baseline models, inspired from a Deep Neural Network (DNN) Nainwal et al. (2022) and an Attention-based CNN-LSTM described in Approach 1Li et al. (2024).

Real-Time System Implementation: We developed a real-time ECG processing system to evaluate how quickly models could make predictions under different heart rate conditions and check if the models meet the deadline for each BPM criteria.

Table 4.2 summarizes the sample sizes and processing time thresholds for three heart rate settings ($High, Mid, Low$) in our real-time system. The "Sample Size" refers to the number of ECG cycles from our database within the certain BPM, while the"Threshold (ms)" denotes the maximum allowable processing time per sample to meet real-time constraints.

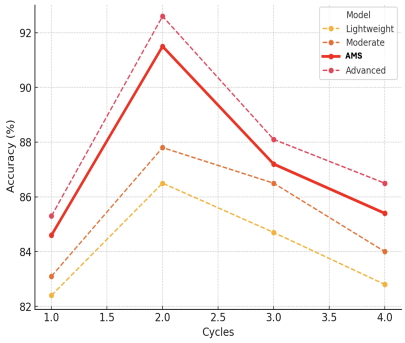
Table 4.2: Settings for Different Heart Rates

Setting	BPM	Sample Size	Threshold (ms)
$High$	$90 \leq$ HR	348,227	1.5
Mid	$70 \leq$ HR< 90	336,238	1.75
Low	HR< 70	336,051	2

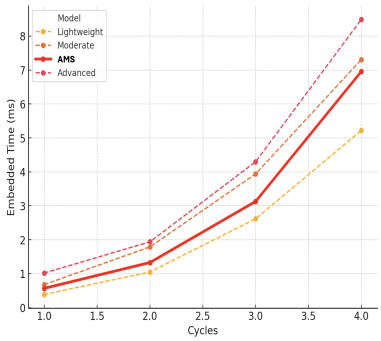
Evaluation Metrics

The performance of the disease detection models was assessed using the following metrics:

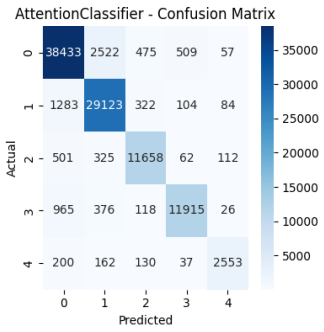
- **Cycles:** The number of ECG cycles provided as input.
- **Accuracy (ACC):** The percentage of correct classifications.



(a) Accuracy Trend Visualization



(b) Inference Trend Visualization



(c) Confusion Matrix

Figure 4.1: Visualization of Accuracy and Inference Trends, and Confusion Matrix for Advanced Model with Two Cycles.

- **Recall (REC):** The proportion of true positives correctly identified out of all actual positives.
- **F1-Score (F1):** The harmonic mean of precision and recall, offering a balanced assessment of model performance.
- **Size (MB):** The model's storage size, indicating complexity and feasibility for deployment on resource-limited devices.
- **Inference (ms):** Average inference time per execution on a Raspberry Pi 4.
- **Deadline Misses (DDL Miss):** The count of missed deadlines from 1,000 data samples.

4.1.2 Experimental Results

Table 4.3 presents the performance metrics of various models across different numbers of ECG cycles. In addition, Figure 4.1 shows the detailed performance for our proposed AMS model. The key findings from the results are as follows:

The **Advanced model** achieved the highest accuracy of **92.6%** with two ECG cycles, indicating its strong ability to learn from additional data. However, its increased model size and inference time at higher cycle counts may limit its suitability for real-time applications on resource-constrained devices. The **Lightweight model** demonstrated commendable performance with minimal computational requirements, achieving **86.5%** accuracy at two cycles. Its small model size and fastest inference times make it well-suited for embedded systems with limited resources. The **AMS model** offered the best balance between accuracy and efficiency,

Table 4.3: Performance Metrics of Model Variants Across Different ECG Cycles

Model	Cycles	ACC (%)	REC (%)	F1 (%)	Size (MB)	Inference (ms)	DDL Miss
DNN	1	55.7	55.4	54.8	0.11	0.05	0
	2	60.9	60.8	60.6	0.29	0.12	0
	3	58.6	56.1	57.2	0.58	1.30	13
	4	57.4	56.0	58.8	1.42	3.41	1000
CNN-LSTM	1	81.1	81.1	81.3	3.25	20.67	1000
	2	87.3	87.2	87.6	5.56	31.33	1000
	3	83.7	83.5	84.1	12.18	59.29	1000
	4	80.4	80.0	79.8	27.51	166.13	1000
SE-ResNet Advanced	1	87.3	87.1	87.0	2.1	1.02	0
	2	**92.6**	**92.5**	**91.1**	6.22	1.94	431
	3	88.1	88.4	88.0	19.5	4.30	1000
	4	86.5	86.8	86.4	44.6	8.49	1000
SE-ResNet Moderate	1	85.1	85.4	85.0	0.49	0.68	0
	2	87.8	88.0	87.7	2.09	1.79	259
	3	86.5	86.8	86.5	5.7	3.94	1000
	4	84.0	84.3	84.7	12.8	7.31	1000
SE-ResNet Lightweight	1	82.4	83.5	83.1	**0.18**	**0.39**	0
	2	86.5	86.0	86.6	0.44	1.05	0
	3	84.7	84.5	83.2	1.8	2.62	1000
	4	82.8	82.0	82.8	4.3	5.22	1000
AMS	1	86.6	86.4	86.8	1.1	0.57	0
	2	91.5	91.0	90.6	4.12	1.33	0
	3	87.2	86.5	87.2	11.6	3.13	1000
	4	85.4	85.0	85.8	33.9	6.96	1000

reaching **91.5%** accuracy with two cycles while maintaining low inference times and zero deadline misses. This makes it ideal for real-time ECG analysis where both promptness and precision are critical.

In contrast, the **DNN model** struggled with low accuracy despite its efficiency, rendering it ineffective for reliable ECG classification. The **CNN-LSTM model**, although achieving high accuracy, faced significantly longer inference times and high deadline miss rates, making it impractical for real-time use on embedded platforms.

CHAPTER

5

DISCUSSION

5.1 Discussion

5.1.1 Model Performance Comparison

Recent deep learning models for ECG classification have achieved very high accuracy on benchmark datasets, often exceeding 98–99% in controlled settings Rajpurkar et al. (2017). For example, various architectures (including RNNs, CNNs, and LSTMs) on the MIT-BIH arrhythmia database report near-perfect performance Rajpurkar et al. (2017). However, such state-of-the-art methods typically use clean and small datasets and involve large and complex networks that are computationally intensive, making them challenging to deploy on resource-constrained devices Lane et al. (2016). In contrast, our proposed approach leverages an anytime neural network architecture that dynamically adjusts inference depth, achieving competitive classification accuracy while offering significant advantages in latency and energy efficiency. Although traditional methods often focus solely on peak accuracy, our design prioritizes real-time responsiveness by allowing early exits when necessary, thereby striking a deliberate balance between computational load and diagnostic precision Teerapittayanon et al. (2016); Zilberstein (1996).

5.1.2 AMS and the Anytime Model

The core of our approach is an Anytime Neural Network augmented with an Adaptive Model Selection that controls the network's execution depth based on the current heart rate. The anytime prediction framework allows the network to produce a valid output at multiple exit points along its depth, providing a trade-off between processing time and prediction accuracy Teerapittayanon et al. (2016). When the heart rate is high—implying a shorter available processing window—the AMS triggers an early exit to ensure that the system produces timely alerts. Conversely, during lower heart rate conditions, the network processes additional layers to deliver a more refined prediction. This HR-driven adaptation enables a dynamic inference budget, which is particularly beneficial in wearable healthcare applications where energy efficiency and responsiveness are critical Zilberstein (1996). Moreover, by incorporating a closed-loop control mechanism, our approach continuously monitors and adjusts the computation depth, ensuring that even under time constraints, the model maintains an acceptable level of accuracy.

5.1.3 Practical Deployment Considerations

Deploying robust ECG classification on wearable devices requires careful consideration of hardware constraints, energy consumption, and system reliability. Wearable and mobile devices typically have limited processing power, memory, and battery life, making the deployment of heavy models impractical Lane et al. (2016). Our adaptive framework addresses these challenges by reducing the average computational load through early exits during periods of high heart rate, thus conserving energy and reducing latency. This design also minimizes heat generation and extends battery life—critical factors for devices that operate continuously. Furthermore, by enabling on-device decision-making, our approach reduces the need for continuous data transmission to remote servers, thereby enhancing both privacy and real-time performance. The integration of an anytime model with AMS also offers scalability; the system can be fine-tuned to meet varying operational requirements, whether for clinical-grade monitoring or consumer wellness applications Mao et al. (2022). In this context, our work represents a significant step toward the deployment of intelligent, energy-efficient cardiac monitoring systems in everyday settings.

5.1.4 Limitations and Future Directions

Despite its advantages, our adaptive model selection framework exhibits some limitations. In high heart rate scenarios, the necessity for early exits may lead to a slight compromise in diagnostic accuracy, as the network might not fully exploit its deeper layers Teerapittayanon et al.

(2016). Additionally, the system's performance can be adversely affected by sensor noise and motion artifacts—common issues in wearable ECG devices—which may necessitate further development of robust pre-processing algorithms Clifford et al. (2006). Real-world implementation also presents challenges, such as precise synchronization of heart rate measurements and the need for extensive validation across diverse populations.

Looking ahead, several improvements are envisioned:

- **Adaptive and Personalized Learning:** Incorporating online learning techniques to adapt the model to individual ECG morphologies over time can enhance diagnostic accuracy. Patient-specific adaptation may be achieved via transfer learning or periodic fine-tuning Mao et al. (2022).
- **Multimodal Sensor Integration:** Fusing ECG data with additional sensor modalities such as accelerometers or photoplethysmography (PPG) could improve robustness, particularly in noisy or motion-intensive environments Zhang et al. (2019); Allen (2007).
- **Enhanced Robustness and Validation:** Implementing consistency training or adversarial training may increase the reliability of intermediate outputs, making the early exits more robust against input perturbations Li et al. (2018).

Overall, our AMS framework with an anytime model presents a viable solution for real-time ECG analysis on wearable devices, balancing computational efficiency and diagnostic accuracy. Continued research in adaptive learning and sensor fusion is essential to further bridge the gap between laboratory performance and real-world application.

CHAPTER

6

CONCLUSIONS

6.1 Conclusion and Future Work

In this study, we presented a novel cardiovascular disease detection model that integrates Residual Blocks, SE layers, and a Global Attention mechanism, further enhanced by an anytime model architecture that supports adaptive model selection. Our experimental results demonstrated that utilizing two ECG cycles significantly improves anomaly detection by reducing noise and artifacts, thereby enabling the model to effectively capture critical diagnostic features. The AMS system, which dynamically chooses between Advanced, Moderate, and Lightweight classifiers based on heart rate thresholds, leverages the anytime model's multiple exit points to balance accuracy and inference speed. This design ensures that rapid predictions are provided during high heart rate conditions while allowing deeper, more accurate processing when time permits. Such efficiency makes our approach particularly suitable for real-time applications in resource-constrained environments, such as wearable health monitoring devices. Future work will focus on enhancing signal preprocessing techniques to further mitigate noise and improve data quality, exploring adaptive BPM thresholding mechanisms to refine model selection accuracy within the AMS framework. Finally, integrating the model into wearable devices and conducting extensive field tests to evaluate its performance in real-world scenarios will provide valuable insights for further refinement and broader clinical adoption.

REFERENCES

Allen, J. (2007). Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, 28(3):R1–R39.

Attia, Z. I., Friedman, P. A., Noseworthy, P. A., et al. (2019). An artificial intelligence-enabled ECG algorithm for the identification of patients with atrial fibrillation during sinus rhythm: A retrospective analysis of outcome prediction. *The Lancet*, 394(10201):861–867.

Benjamin, E. J., Virani, S. S., Callaway, C. W., et al. (2019). Heart disease and stroke statistics—2019 update: a report from the american heart association. *Circulation*, 139(10):e56–e528.

Buttazzo, G. C. (2011). *Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications*, volume 24. Springer Science & Business Media.

Cai, H., Gan, C., and Han, S. (2019). Once-for-all: Train one network and specialize it for efficient deployment. *arXiv preprint arXiv:1908.09791*.

Carreiras, C., Alves, A. P., Lourenço, A., Canento, F., Silva, H., Fred, A., et al. (2015–). BioSPPy: Biosignal processing in Python. [Online; accessed <today>].

Clifford, G. D., Azuaje, F., and McSharry, P. E. (2006). *Advanced Methods and Tools for ECG Data Analysis*. Artech House.

Elgendi, M. (2013). Fast qrs detection with an optimized knowledge-based method: Evaluation on 11 standard ecg databases. *PLOS ONE*, 8(9):e73557.

Galli, A., Ambrosini, F., and Lombardi, F. (2016). Holter monitoring and loop recorders: From research to clinical practice. *Arrhythmia & Electrophysiology Review*, 5(2):136–141.

Guo, Z. and Baruah, S. K. (2015). Uniprocessor edf scheduling of avr task systems. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*, ICCPS '15, page 159–168, New York, NY, USA. Association for Computing Machinery.

Gupta, U., Paluru, N., Nankani, D., Kulkarni, K., and Awasthi, N. (2024). A comprehensive review on efficient artificial intelligence models for classification of abnormal cardiac rhythms using electrocardiograms. *Heliyon*, 10(5):e26787.

Hamilton, P. (2002). Open source ecg analysis. In *Computers in Cardiology*, pages 101–104.

Han, S., Mao, H., and Dally, W. J. (2016). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In *4th International Conference on Learning Representations (ICLR)*.

Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., and Ng, A. Y. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature Medicine*, 25(1):65–69.

Islam, M. R., Qaraqe, M., Qaraqe, K., and Serpedin, E. (2024). Cat-net: Convolution, attention, and transformer based network for single-lead ecg arrhythmia classification. *Biomedical Signal Processing and Control*, 93:106211.

Kleiger, R. E. and Stein, P. K. (2005). Heart rate variability: measurement and clinical utility. *Annals of Noninvasive Electrocardiology*, 10(1):88–101.

Lane, N. D., Bhattacharya, S., and Georgiev, P. (2016). Deepx: A resource-efficient deep learning framework for mobile sensing. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1–10. ACM.

Li, Y. et al. (2018). Improving early exits in multi-exit networks with consistency training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 120–128.

Li, Y., Sui, N., Gehi, A., Guo, C., and Guo, Z. (2024). Cardiacrt-nn: Real-time detection of cardiovascular disease using self-attention cnn-lstm for embedded systems. In Le, X. and Zhang, Z., editors, *Advances in Neural Networks – ISNN 2024*, pages 610–621, Singapore. Springer Nature Singapore.

Liu, C., Li, Y., Li, C., Zhang, X., and Liu, C. (2021). ECG signal denoising and baseline wander correction based on CEEMDAN and wavelet threshold. *Sensors*, 21(5):1572.

Liu, C. L. and Layland, J. W. (1973). Scheduling algorithms for multiprogramming in a hard-real-time environment. *Journal of the ACM (JACM)*, 20(1):46–61.

Mao, H. et al. (2022). A low-energy neural network accelerator for always-on ecg monitoring. In *Proceedings of the IEEE International Conference on Embedded Systems*, pages 220–227.

Mohamed Suhail, M. and Abdul Razak, T. (2022). Cardiac disease detection from ecg signal using discrete wavelet transform with machine learning method. *Diabetes research and clinical practice*, 187:109852–109852.

Nainwal, A., Kumar, Y., and Jha, B. (2022). An ecg classification using dnn classifier with modified pigeon inspired optimizer. *Multimedia tools and applications*, 81(7):9131–9150.

Organization, W. H. (2021). Cardiovascular diseases (cvds). *WHO*. `https://www.who.int/health-topics/cardiovascular-diseases`.

Pan, J. and Tompkins, W. J. (1985). A real-time qrs detection algorithm. *IEEE Transactions on Biomedical Engineering*, BME-32(3):230–236.

Patel, S., Park, H., Bonato, P., Chan, L., and Rodgers, M. (2012). A review of wearable sensors and systems with application in rehabilitation. *Journal of NeuroEngineering and Rehabilitation*, 9(1):21.

Rajpurkar, P., Hannun, A. Y., Haghpanahi, M., Bourn, C., and Ng, A. Y. (2017). Cardiologist-level arrhythmia detection with convolutional neural networks. *Nature Medicine*, 23(1):43–49.

Research, G. V. (2019). Wearable medical devices market size, share & trends analysis report. *Grand View Research*. https://www.grandviewresearch.com/industry-analysis/wearable-medical-devices-market.

Reyna, M., Sadr, N., Perez Alday, E., Gu, A., Shah, A., Robichaux, C., Rad, A., Elola, A., Seyedi, S., Ansari, S., et al. (2021). Will two do? varying dimensions in electrocardiography: the physionet/computing in cardiology challenge 2021. In *2021 Computing in Cardiology (CinC)*, pages 1–4. IEEE.

Ribeiro, A. L. P., Ribeiro, A. L., et al. (2020). Tele-electrocardiography and big data: The code (r) evolution. *Arquivos Brasileiros de Cardiologia*, 115(5):774–776.

Strodthoff, N., Wagner, P., Schaeffter, T., and Samek, W. (2021). Deep learning for ECG analysis: Benchmarks and insights from ptb-xl. *IEEE Journal of Biomedical and Health Informatics*, 25(5):1519–1528.

Teerapittayanon, S., McDanel, B., and Kung, H. T. (2016). Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2464–2469. IEEE.

Zhang, Z., Li, M., and Wang, X. (2019). A wearable system for robust ecg monitoring using integrated accelerometer data. *IEEE Sensors Journal*, 19(15):6001–6008.

Zilberstein, S. (1996). Using anytime algorithms in intelligent systems. *AI Magazine*, 17(3):73–83.

APPENDIX

APPENDIX

A

DIAGNOSIS CODES

A summary of all disease labels and severity levels is documented in Table A.1.

Table A.1: Diagnosis Codes by Seriousness

Dx	**Seriousness**	**SNOMEDCTCode**	**Total**
1st degree av block	0	270492004	3534
sinus rhythm	0	426783006	28971
premature atrial contraction	0	284470004	3041
prolonged pr interval	0	164947007	392
sinus arrhythmia	0	427393009	3790
supraventricular premature beats	0	63593006	224
atrial bigeminy	0	251173003	6
atrial rhythm	0	106068003	215
junctional premature complex	0	251164006	13
sinus atrium to atrial wandering rhythm	0	17366009	7
supraventricular bigeminy	0	251168009	1
wandering atrial pacemaker	0	195101003	9

Continued on next page

Table A.1 – Continued

Dx	Seriousness	SNOMEDCTCode	Total
pacing rhythm	1	10370003	1481
premature ventricular contractions	1	427172004	1279
sinus bradycardia	1	426177001	18918
sinus tachycardia	1	427084000	9657
ventricular premature beats	1	17338001	659
atrial pacing pattern	1	251268003	52
blocked premature atrial contraction	1	251170000	67
mobitz type i wenckebach atrioventricular block	1	54016002	34
decreased qt interval	1	77867006	3
ventricular bigeminy	1	11157007	101
ventricular ectopics	1	164884008	1944
ventricular escape beat	1	75532003	60
ventricular pacing pattern	1	251266004	46
ventricular trigeminy	1	251180001	37
atrial fibrillation	2	164889003	5255
atrial flutter	2	164890007	8374
bradycardia	2	426627000	295
accelerated idioventricular rhythm	2	61277005	14
accelerated junctional rhythm	2	426664006	31
atrial fibrillation and flutter	2	195080001	41
atrial tachycardia	2	713422000	340
atrioventricular junctional rhythm	2	29320008	145
atrioventricular node reentrant tachycardia	2	251166008	16
atrioventricular reentrant tachycardia	2	233897008	26
brady tachy syndrome	2	74615001	2
chronic atrial fibrillation	2	426749004	1
cardiac dysrhythmia	2	698247007	16
congenital incomplete atrioventricular heart block	2	204384007	2
idioventricular rhythm	2	49260003	2
junctional tachycardia	2	426648003	30
paroxysmal atrial fibrillation	2	282825002	2
paroxysmal supraventricular tachycardia	2	67198005	27

Continued on next page

Table A.1 – Continued

Dx	Seriousness	SNOMEDCTCode	Total
sinoatrial block	2	65778007	14
sinus node dysfunction	2	60423000	2
shortened pr interval	2	49578007	28
supraventricular tachycardia	2	426761007	787
u wave abnormal	2	164937009	137
ventricular pre excitation	2	195060002	20
wolff parkinson white pattern	2	74390002	160
prolonged qt interval	3	111975006	1907
av block	3	233917008	323
2nd degree av block	3	195042002	124
junctional escape	3	426995002	84
myocardial infarction	3	164865005	6144
myocardial ischemia	3	164861001	2559
rapid atrial fibrillation	3	314208002	2
s t changes	3	55930002	5009
st depression	3	429622005	3645
acute myocardial infarction	4	57054005	55
acute myocardial ischemia	4	413444003	2
atrioventricular dissociation	4	50799005	59
complete heart block	4	27885002	127
mobitz type II atrioventricular block	4	426183003	7
paroxysmal ventricular tachycardia	4	425856008	124
sinus arrest	4	5609005	33
st elevation	4	164931005	628
ventricular escape rhythm	4	81898007	98
ventricular fibrillation	4	164896001	97
ventricular flutter	4	111288001	8
ventricular tachycardia	4	164895002	12