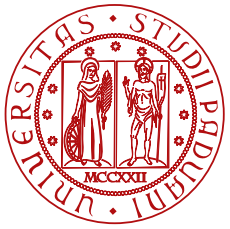


Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione

Master Thesis in ICT for Internet and Multimedia

A contrastive learning approach for time series with application to anomaly detection in healthcare

Master Candidate

Griselda Kolici

Student ID 2071532

Supervisor

Prof. Roberto Corvaja

University of Padova

Co-supervisor

Prof. Giulia Cisotto

Dr. Alberto Zancanaro

Academic Year: 2023/2024

Graduation Date: December 4, 2024

To my family

Abstract

Time-series analysis, particularly in biomedical signals like Electrocardiograms (ECG), poses the critical challenge of distinguishing between global patterns shared across subjects and individual-specific features unique to each patient. Many existing methods fail to achieve this balance, limiting their applicability to tasks requiring both generalization and fine-grained differentiation.

This thesis addresses this limitation by exploring the central research question: *How can contrastive learning techniques help distinguish general patterns from individual fingerprints in time-series data?* To answer this, a contrastive learning framework is proposed, leveraging advanced preprocessing and deep learning techniques. Seasonal-Trend decomposition using Loess (STL) is employed to preprocess ECG signals, isolating periodic components while preserving critical details. The proposed Siamese network architecture, enhanced with Residual Blocks and Squeeze-and-Excitation (SE) mechanisms, generates robust embeddings that effectively balance global pattern extraction and individual fingerprint preservation.

The research question is evaluated through contrastive loss, t-SNE visualizations of the embedding space, and L2 distance distributions for positive and negative pairs, which together assess the networks ability to capture intra-subject consistency and inter-subject variability. Anomaly detection is employed as a representative use case to further demonstrate the frameworks potential, with performance assessed using metrics such as accuracy, precision, recall, and AUC.

While the focus of this work lies in addressing the research question, the methodologies and insights extend beyond the specific case of ECG anomaly detection. By bridging the gap between global pattern extraction and individual fingerprint preservation, this thesis provides a robust foundation for applying contrastive learning techniques to a wide range of time-series tasks.

Contents

List of Figures

List of Tables

List of Algorithms

List of Code Snippets

List of Acronyms

AI Artificial Intelligence

CNN Convolutional Neural Network

DL Deep Learning

ECG Electrocardiogram

MIT-BIH Massachusetts Institute of Technology - Beth Israel Hospital

SE Block Squeeze-and-Excitation Block

STL Seasonal-Trend decomposition using LOESS

BCE Binary Cross-Entropy

SNN Siamese Neural Network

F1-Score Harmonic mean of precision and recall

CL Contrastive Learning

QRS QRS Complex (a waveform in ECG representing ventricular depolarization)

PR PR Interval (time from atrial depolarization to ventricular depolarization in ECG)

ST ST Segment (flat section of ECG between ventricular depolarization and repolarization)

EEG Electroencephalogram

CLECG Contrastive Learning for Electrocardiograms

CLOCS Contrastive Learning of Cardiac Signals Across Space, Time, and Patients

LSTM Long Short-Term Memory Network

RNN Recurrent Neural Network

VAE Variational Autoencoder

GAN Generative Adversarial Network

t-SNE t-distributed Stochastic Neighbor Embedding

3D Three-Dimensional

1 Introduction

1.1 Introduction

The analysis of time-series data, particularly in medical contexts such as Electrocardiograms (ECG), presents the dual challenge of extracting general patterns while preserving subject-specific characteristics [27] [40] [1]. Global patterns, like the QRS complex, are common across individuals and essential for tasks like arrhythmia detection, whereas subtle variations in waveform amplitude, morphology, or rhythm serve as unique "fingerprints" for each patient.

Despite their diagnostic importance, existing approaches often struggle to balance these requirements, leading to limited applicability in tasks that demand both generalization across subjects and fine-grained differentiation. This limitation is particularly pronounced in applications such as anomaly detection or personalized classification, where the ability to disentangle global patterns from individual-specific features is crucial [33].

This thesis addresses these limitations by investigating the central research question: *How can contrastive learning techniques help to distinguish general patterns from individual fingerprints in time-series data?* To explore this question, ECG signal analysis serves as a representative task, given its clinical importance and the availability of annotated datasets.

The methodology involves leveraging Seasonal-Trend decomposition using Loess (STL) for preprocessing and a Siamese network architecture with residual connections and Squeeze-and-Excitation (SE) blocks for learning discriminative embeddings. STL is used to isolate the periodic and residual components of

ECG signals, enabling a structured approach to preprocess time-series data. The Siamese network is tailored to maximize inter-subject variability while retaining intra-subject consistency.

This work extends the existing literature by introducing a contrastive learning framework tailored to ECG signals that combines robust preprocessing and advanced neural network design. The role of embeddings in distinguishing subject-specific characteristics from general patterns in time-series data is explored. Additionally, the impact of seasonal parameter tuning, signal reconstruction, and embedding visualization is evaluated.

The structure of this thesis is as follows. Chapter 2 provides the background and reviews the state of the art in contrastive learning for time-series data. Chapter 3 details the proposed methodology, including the data preprocessing techniques, pair creation strategies, network architecture and adaption to the anomaly detection task. Chapter 4 presents the results of the experiments, analyzing the embeddings, distance distributions, and downstream performance metrics. Finally, Chapter 5 concludes with a discussion of the findings, their implications, and future research directions.

By addressing the outlined research question, this thesis aims to contribute to a deeper understanding of how contrastive learning can disentangle global and individual-specific patterns in time-series data, ultimately advancing methods for tasks like anomaly detection and beyond.

2

Background and related work

2.1 Overview of the heart and ECG

The heart is a vital organ responsible for circulating blood throughout the body, ensuring the delivery of oxygen and nutrients while removing waste products such as carbon dioxide. This process is driven by the rhythmic contractions of the heart, which are regulated by its intrinsic electrical conduction system. The heart consists of four chambers: the right and left atria and the right and left ventricles, which work in a coordinated manner to pump blood through the pulmonary and systemic circulations. The electrical activity of the heart, generated by the sinoatrial (SA) node-the natural pacemaker-propagates through specialized conduction pathways, including the atrioventricular (AV) node, bundle of His, and Purkinje fibers[35]. This electrical activity ensures synchronized contractions, enabling efficient blood flow throughout the body.

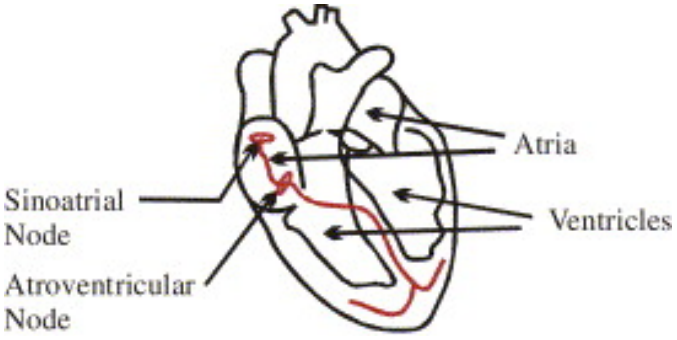


Figure 2.1: Illustration of the heart's conduction system, including the sinoatrial node and atrioventricular node, adapted from Israel et al. (2005) [16].

Electrocardiography (ECG) is a widely used non-invasive diagnostic technique for recording the heart's electrical activity over time. By placing electrodes on the body, typically on the chest, arms, and legs, the electrical impulses generated by the heart can be captured and displayed as a series of waveforms. These waveforms provide valuable information about the heart's rhythm, conduction pathways, and overall function. A standard 12-lead ECG records the heart's electrical activity from multiple angles, offering a comprehensive view of its function [24].

The ECG waveform consists of several characteristic components that correspond to specific phases of the cardiac cycle. The P wave represents atrial depolarization, marking the beginning of the cardiac cycle. The QRS complex, the most prominent feature of the ECG, reflects ventricular depolarization, which leads to ventricular contraction. Following the QRS complex, the T wave represents ventricular repolarization, or the recovery phase. Additional intervals and segments, such as the PR interval and ST segment, provide further insights into the timing and coordination of electrical events within the heart [36]. Changes in these deflections might occur due to abnormal heart conditions and also show signs of inconsistency based on individual patients [42].

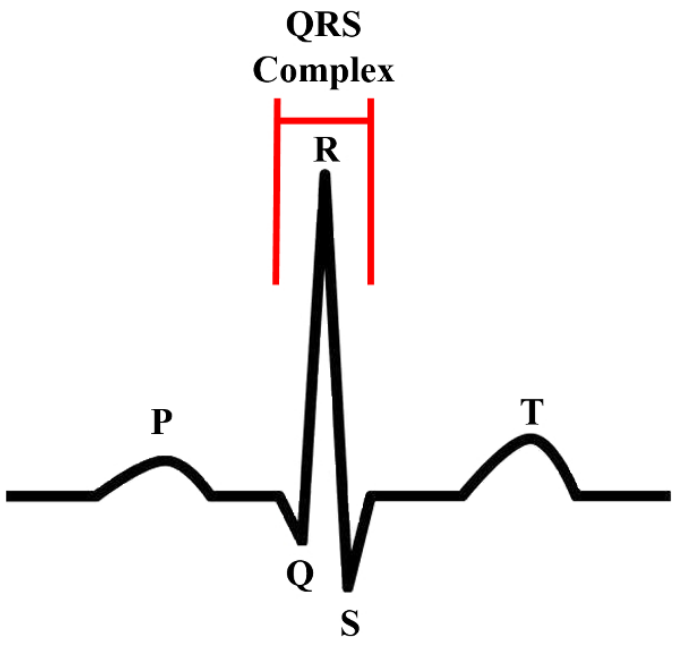


Figure 2.2: Typical ECG waveform showing key components such as the P wave, QRS complex, and T wave. Adapted from Anbalagan et al. (2023) [3].

A normal ECG signal exhibits predictable patterns, with quasi-periodic in-

tervals between beats and consistent wave shapes. It serves as a baseline for identifying deviations that may signal abnormalities. Abnormal ECG signals, on the other hand, are characterized by changes in waveform morphology, duration, or timing.

The electrocardiogram (ECG) is an essential tool for diagnosing heart disease, with computer-aided systems improving diagnostic accuracy and reducing healthcare costs [19]. Automated ECG analysis systems aim to assist clinicians by detecting abnormalities with high accuracy, even in large datasets. However, this task is complicated by the inherent variability of ECG signals across individuals [45][19]. While some variations are indicative of pathological conditions, others may simply reflect normal differences between individuals. As a result, robust methods are required to distinguish global patterns that are common across populations from individual-specific "fingerprints" unique to each patient. This capability is especially important in anomaly detection, where the goal is to identify subtle deviations that may signal a pathological state.

In this thesis, anomaly detection in ECG signals serves as a testbench for evaluating the proposed contrastive learning framework. While the primary objective of the research is to develop a method that can effectively distinguish general patterns from individual-specific features in time-series data, ECG signals provide an ideal dataset for testing these capabilities due to their complexity, quasi-periodic behavior, and clinical relevance. By focusing on ECG anomaly detection, this study demonstrates the potential of contrastive learning to address real-world challenges in time-series analysis, particularly in healthcare applications.

2.2 Deep learning for time-series analysis

Deep learning, a subset of machine learning, has revolutionized the field of time-series analysis by enabling models to automatically learn hierarchical and temporal features from raw data. Unlike traditional methods, which often rely on handcrafted feature extraction, deep learning methods leverage neural networks with multiple layers to capture complex patterns and dependencies within time-series data [31]. This capability has made deep learning an indispensable tool for a wide range of applications, particularly in the healthcare domain.

In the healthcare industry, deep learning has demonstrated significant po-

tential for improving the accuracy and efficiency of predictive models [2]. Time-series data, such as electrocardiograms (ECGs), exhibit temporal patterns that are well-suited for analysis by deep learning architectures like Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks. For instance, CNNs excel at capturing local patterns and hierarchical structures, while RNNs and LSTMs are designed to handle sequential dependencies and long-term temporal correlations [14]. These architectures have been successfully employed in various healthcare applications, including disease prediction, anomaly detection, and personalized medicine [32].

A recent study highlights the critical role of deep learning in time-series prediction for healthcare, emphasizing its ability to handle large volumes of high-dimensional data [31]. Despite its advantages, challenges such as model interpretability, computational cost, and the need for large annotated datasets remain prevalent [2]. Furthermore, integrating deep learning into IoT-enabled smart healthcare systems has opened new frontiers for real-time monitoring and predictive analytics, but it also introduces issues related to scalability and data security [32].

Deep learning models are often categorized as either generative or discriminative. Generative models, such as Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs), are well-suited at capturing the underlying distribution of data and generating synthetic samples. Discriminative models, including CNNs and LSTMs, focus on learning decision boundaries for specific tasks, such as classification or regression [14]. These approaches have proven effective in advancing time-series prediction and anomaly detection, particularly in biological system modeling and healthcare applications.

2.3 State of the Art

Contrastive learning (CL) has become an essential method in self-supervised learning, particularly for extracting meaningful representations from unlabeled data [21]. The foundation of contrastive learning lies in distinguishing between positive pairs (similar signals) and negative pairs (dissimilar signals) in a shared embedding space. A detailed review by Jaiswal et al. [18] provides an overview of its development, highlighting how CL enables models to identify patterns and representations without the need for labeled data. Chen et al. [6] introduced

SimCLR, a notable contrastive learning framework that uses data augmentations to create positive and negative pairs, establishing a baseline for many subsequent studies in the field.

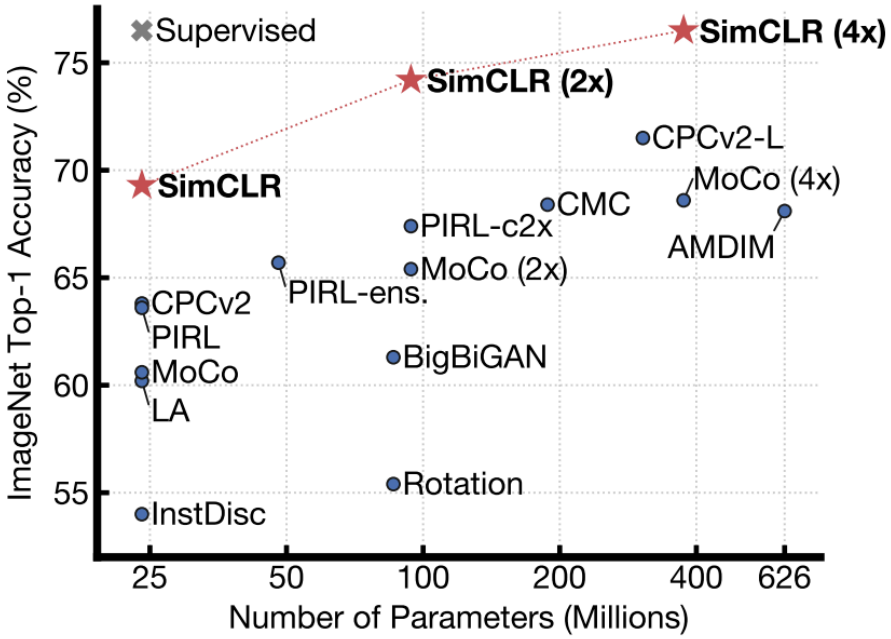


Figure 2.3: ImageNet Top-1 accuracy of linear classifiers trained on representations learned with different self-supervised methods (pretrained on ImageNet). The gray cross indicates a supervised ResNet-50 baseline. SimCLR, highlighted in bold, demonstrated significant improvements in accuracy. Source: Chen et al. [6].

In recent years, the application of contrastive learning has extended into time-series data, particularly in medical domains such as electrocardiogram (ECG) and electroencephalogram (EEG) analysis. A systematic review by Liu et al. [26] highlights how contrastive self-supervised learning has shown success in tasks such as anomaly detection, patient monitoring, and disease classification. In this context, CL techniques have proven effective for overcoming the scarcity of labeled data by learning robust and generalizable embeddings [7].

For ECG analysis, frameworks like CLECG [5] and CLOCS [20] demonstrate how contrastive learning can handle inter-subject variability and classify arrhythmias with improved accuracy. These methods often rely on augmentations specific to time-series signals (see Figure 2.4), such as random cropping, time warping, and noise injection, to generate diverse and meaningful representations [26].

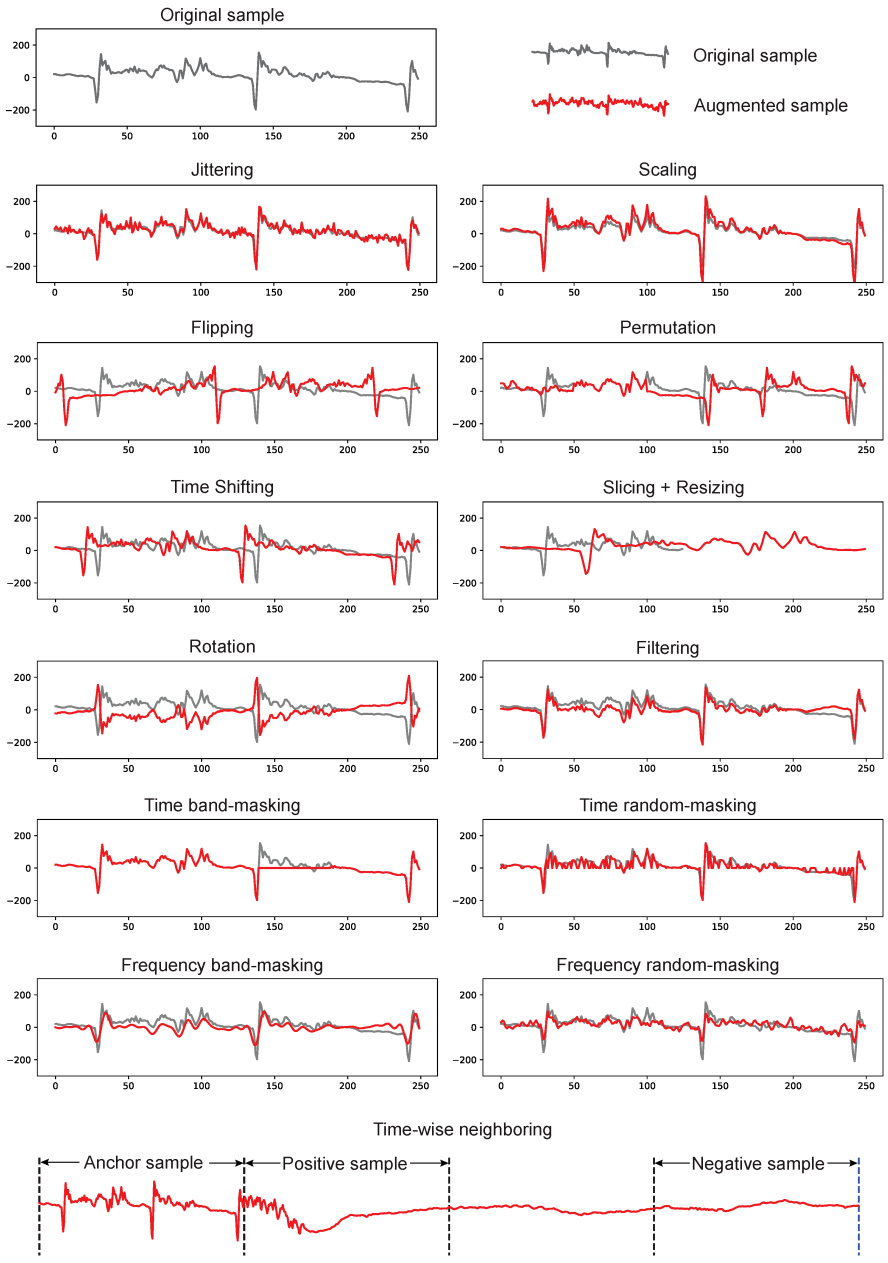


Figure 2.4: Illustration of various data augmentations applied to time-series data, adapted from Liu et al. [26]

Another study by Rabbani et al. [38] applies contrastive self-supervised learning to detect stress in ECG signals, further showcasing the flexibility of CL in capturing subtle patterns related to physiological conditions. Similarly, Lee et al. [22] utilized contrastive techniques for anomaly detection in time-series data, emphasizing ECG as a domain where CL methods excel at identifying rare and complex abnormalities.

In the domain of EEG, contrastive learning has been employed to address diverse challenges, such as emotion recognition and motor imagery classification. For example, Abbas et al. [13] used semi-supervised contrastive learning for generalizable EEG classification, demonstrating its ability to learn invariant features across subjects. Shen et al. [39] further explored subject-invariant EEG representations for cross-subject emotion recognition, underscoring how CL can address inter-subject variability while preserving individual-specific details. Additionally, Liu et al. [25] extended CL to semi-supervised time-series classification, leveraging unlabeled EEG data to improve the performance of models on labeled subsets.

Contrastive learning's impact on healthcare applications goes beyond EEG and ECG. For instance, Matton et al. [29] introduced a novel framework to apply contrastive learning to electrodermal activity (EDA) signals, demonstrating its effectiveness in stress detection. By learning robust representations of EDA signals, their framework enabled better recognition of stress levels across individuals, addressing challenges of inter-subject variability. Similarly, Tang et al. [43] explored the use of contrastive learning for human activity recognition in healthcare, showing how self-supervised techniques can improve models' ability to generalize across diverse activity datasets. These examples highlight the versatility of contrastive learning in analyzing various biosignals, paving the way for advancements in patient monitoring, personalized medicine, and diagnostic tools beyond traditional ECG and EEG applications.

Despite the remarkable advancements, there are still gaps in existing methods. Many approaches struggle to disentangle global patterns shared across subjects from individual-specific features unique to each patient. This challenge is particularly evident in tasks like anomaly detection, where models must detect rare and subtle abnormalities while accounting for the inherent variability in ECG signals. Moreover, the effectiveness of CL methods heavily depends on the design of augmentations [26]. Improper transformations can distort temporal dependencies or introduce artifacts, potentially degrading the quality of learned representations. These challenges highlight the need for more sophisticated techniques that can capture the interplay between shared and individual-specific characteristics in time-series data.

This thesis aims to address these limitations by exploring the central research question: *How can contrastive learning techniques help to distinguish general patterns from individual fingerprints on time-series?* To answer this question, the

work focuses on developing a contrastive learning framework tailored to ECG anomaly detection, using it as a representative task. By leveraging techniques such as Seasonal-Trend decomposition using Loess (STL) for preprocessing and a Siamese network architecture with Residual Blocks that include Squeeze-and-Excitation (SE) blocks, the proposed approach seeks to balance the extraction of global patterns and the preservation of individual-specific features. While this thesis specifically investigates ECG anomaly detection, the insights gained from addressing the research question can be adapted to a wide range of downstream tasks in time-series analysis.

In summary, contrastive learning has emerged as a powerful tool for self-supervised representation learning, with applications spanning multiple domains, including healthcare. Existing studies demonstrate its effectiveness in ECG and EEG analysis, highlighting its potential for tasks like anomaly detection, classification, and emotion recognition. However, the challenges of disentangling global and individual-specific features and designing effective augmentations remain open problems. By addressing these issues, this thesis contributes to advancing the state of the art in contrastive learning for time-series, with a specific focus on healthcare applications.

3 Methodology

This chapter details the methodological framework developed to address the research question. The proposed approach involves preprocessing raw ECG signals, leveraging Seasonal-Trend decomposition to extract meaningful components, training a Siamese network using contrastive loss, and adapting the model to an anomaly detection use case. The steps described below lay the foundation for analyzing time-series data with an emphasis on distinguishing individual-specific patterns.

3.1 Dataset

3.1.1 Description of the MIT-BIH Arrhythmia Database

The MIT-BIH Arrhythmia Database [12] is a widely recognized resource in electrocardiogram (ECG) research, offering a diverse collection of annotated recordings. The dataset contains 48 recordings from 47 subjects, each sampled at 360 Hz with a duration of approximately 30 minutes. Two ECG leads are available for each recording, capturing the heart's electrical activity from different perspectives.

Each recording is accompanied by detailed annotations provided by expert cardiologists. These annotations indicate the occurrence of specific cardiac events, such as normal beats (denoted by 'N'), premature ventricular contractions ('V'), and atrial fibrillation episodes, among others. The annotations are aligned with the corresponding sample indices, enabling precise localization of

cardiac events within the signal. This detailed labeling supports the development and evaluation of supervised learning models by serving as ground truth for arrhythmia detection and classification tasks.

This dataset is well-suited for training and evaluating anomaly detection models due to its inclusion of a wide range of arrhythmias and normal beats. Furthermore, the database has played an important role in stimulating the development of arrhythmia analyzers and promoting the use of common databases for medical device evaluation and research [30].

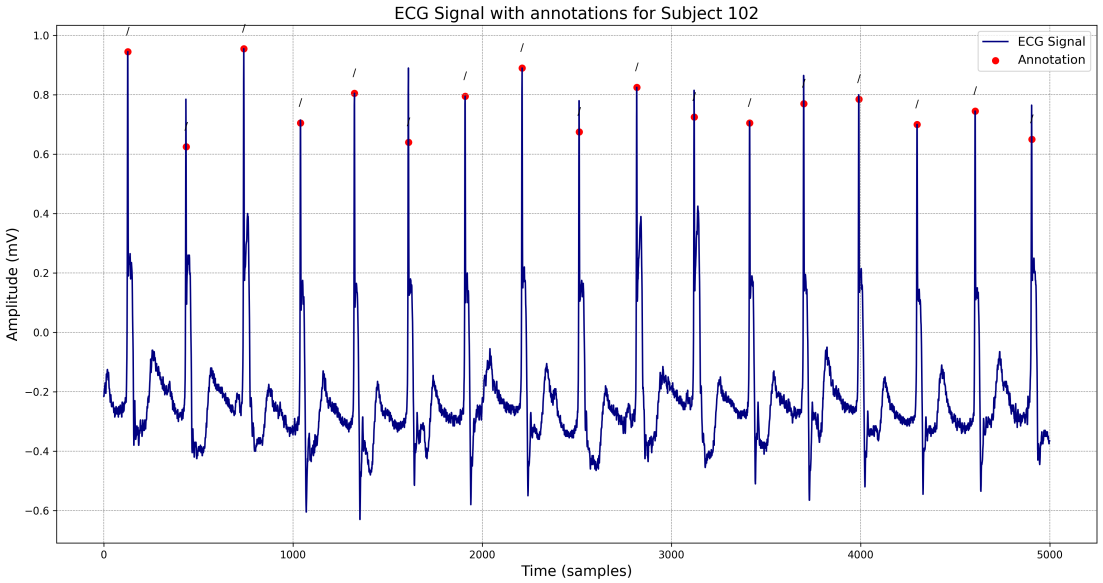


Figure 3.1: An example of an ECG signal segment (5000 samples) from Subject 102 in the MIT-BIH Arrhythmia Database. The red markers indicate annotations corresponding to different cardiac events, such as arrhythmias.

3.1.2 Preprocessing

To make the raw ECG signals suitable for analysis and subsequent processing, several steps were carefully followed. These steps aimed to simplify the data while preserving its essential features and addressing variations that could affect the results.

The first step was to select a single channel (lead) from each recording. This decision was primarily made to reduce the complexity of the data while still retaining enough diagnostic information. The choice to use only one lead was further supported by findings from Mathews et al. (2018) [28] and Gadaleta et

al [10], who demonstrated the viability of single-lead ECG analysis for classification tasks. Their work demonstrated that focusing on a single lead, could still yield reliable results. Building on this validation, I decided to adopt this approach for my analysis.

Next, the selected signal was normalized. Normalization was performed using Min-Max scaling, which adjusts the amplitude of the signal to a range between 0 and 1. This step was essential for ensuring that all signals were on the same scale, reducing variations caused by differences in recording equipment.

Signal decomposition and Reconstruction

In traditional time-series data augmentation, methods such as time shifting, scaling, and adding noise are frequently employed to create diverse representations of signals. However, these approaches often fail to capture the intricate structure of the data, especially in physiological signals like ECG, where distinguishing general patterns from individual-specific features is critical. To overcome these limitations, I utilized Seasonal-Trend decomposition using LOESS (STL decomposition), a powerful statistical tool introduced by Cleveland et al. [9].

STL decomposition is a robust technique for decomposing time-series data into three distinct components:

- **Trend Component:** This represents the long-term behavior of the signal, capturing gradual changes over time. It reflects underlying patterns such as baseline shifts in ECG signals.
- **Seasonal Component:** This captures repeating short-term fluctuations in the data, such as periodic variations that occur at fixed intervals.
- **Residual (Noise) Component:** This accounts for the irregular variations or noise that cannot be explained by the trend or seasonal components.

The application of STL decomposition allowed me to isolate and analyze these components, providing a structured representation of the ECG signals. By focusing on the residual component, I was able to investigate and highlight the "fingerprint" of each individual signal, while the trend and seasonal components captured the broader, general patterns.

This approach aligns with insights from Lee et al. [23], who leveraged STL decomposition within contrastive learning frameworks for time-series anomaly detection. In my study, the decomposed components were utilized for data augmentation, creating positive pairs.

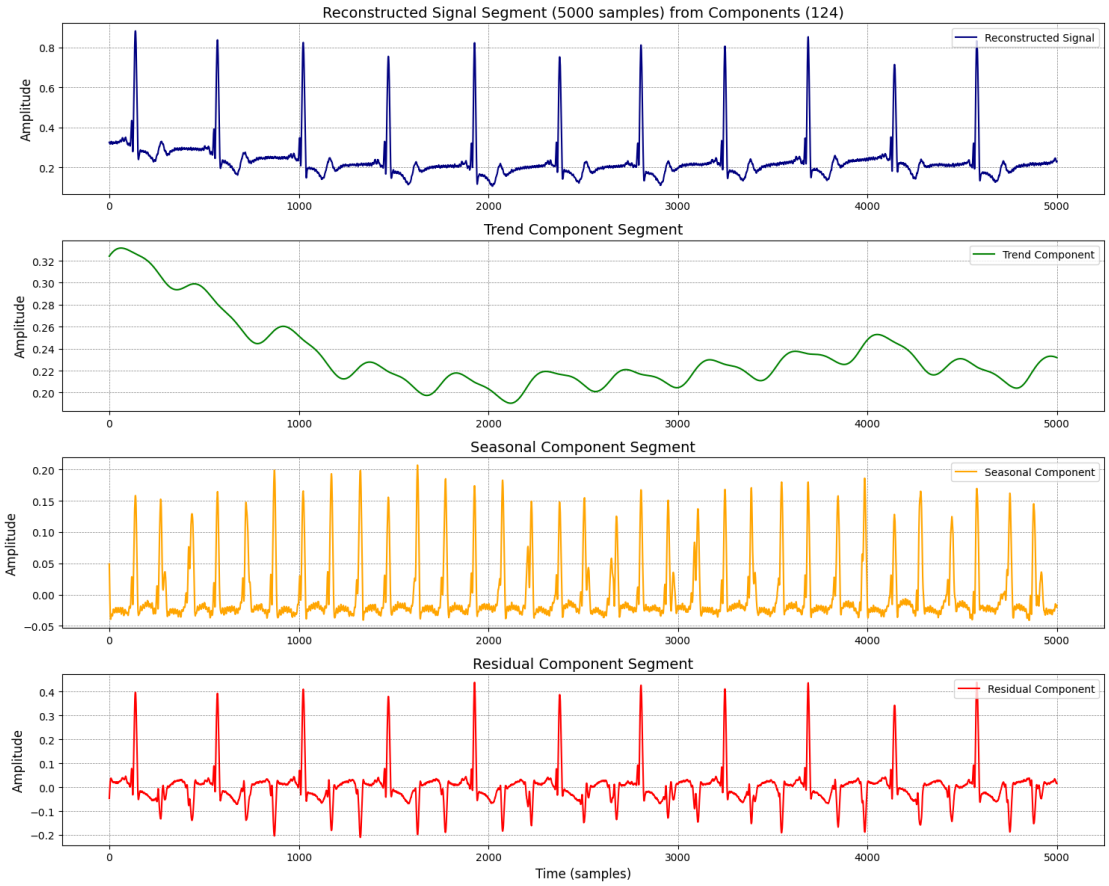


Figure 3.2: An example of a normalized ECG signal segment (5000 samples) from Subject 124 in the MIT-BIH Arrhythmia Database, followed by its decomposed components: trend, seasonal, and residual.

Reconstruction of the signals using 1D CNN

Based on the work by Lee et al. (2023) [23], I decided to use a one-dimensional Convolutional Neural Network (1D CNN) to reconstruct ECG signals from their decomposed components generated by Seasonal-Trend decomposition (STL). The goal of the reconstruction process is to combine these components into a single-channel signal that retains the essential features of the original ECG.

The 1D CNN is designed to process these three input channels independently before combining their outputs into a unified representation. The trend and seasonal components, which represent stable and recurring patterns, are passed through a shared convolutional block to capture general and rhythmic trends. The residual component, which contains irregular variations and individual-specific characteristics, is processed separately through its own convolutional

block. To emphasize its importance, the residual is weighted more heavily during the reconstruction process, as these features may contain subject-specific fingerprints.

The outputs from the trend-seasonal and residual blocks are concatenated, and additional convolutional and upsampling layers are applied to merge the extracted features and reconstruct the signal to its original length.

To create input data for the model, a sliding window approach with overlapping segments was employed. Each window contained 5000 samples, corresponding to approximately 13.9 seconds of data at a sampling frequency of 360 Hz (360 samples per second). With a 50% overlap (2500 samples), this method ensured the preservation of local signal patterns, as the high sampling frequency relative to the window size was sufficient to capture the essential features of the ECG signal. After processing, the reconstructed signal was reassembled by averaging overlapping windows. This way, the reconstructed signals provide a simplified and consistent representation of the ECG while retaining critical information from the original signal.

3.2 Pair Creation for Contrastive Learning

To prepare the data for training the Siamese network, positive and negative pairs were generated to represent intra-subject consistency and inter-subject variability, respectively. These pairs are essential for the contrastive learning approach, as they guide the model to distinguish between general patterns and subject-specific features.

Positive Pairs: Positive pairs were constructed by pairing the original ECG signal with its corresponding reconstructed signal obtained after STL decomposition and 1D CNN processing. Both signals in a positive pair belong to the same subject, with the reconstructed signal retaining the subject-specific characteristics while incorporating slight variations due to the decomposition and reconstruction processes. This pairing strategy emphasizes intra-subject consistency, training the model to recognize a unique "fingerprint" of each subject despite minor variations introduced by preprocessing.

Negative Pairs: Negative pairs were created by pairing the original ECG signal of one subject with the original signal of a different subject. For consistency,

the same lead was selected for both subjects to ensure comparability across the signals. This inter-subject pairing simulates the task of distinguishing between distinct individuals, encouraging the model to identify individual-specific features as opposed to more general, shared patterns. By comparing signals from different subjects, the model learns to focus on discriminative features unique to each subject. I did not apply Seasonal-Trend decomposition (STL) in this step, as the focus was on working directly with the original signals.

This positive and negative pairing strategy provides a structured framework for the contrastive learning objective, aligning with the broader research goal of distinguishing general patterns from individual-specific fingerprints in time-series data.

3.3 Model architecture and training

3.3.1 Introduction to the Siamese Network

The Siamese network, a neural network architecture designed for pairwise comparisons between inputs, plays a central role in this study's novel approach. By leveraging contrastive learning, the network was employed to analyze ECG signals, specifically addressing the challenge of distinguishing general patterns shared across subjects from unique individual-specific fingerprints. This methodology represents an innovative solution to the research question. Each branch of the Siamese network processes one of the input signals in a pair, and the network learns a similarity measure based on the embeddings of the input signals.

Inspired by previous studies, such as Vasconcellos et al. (2023) [44], which demonstrated the effectiveness of Siamese networks in heartbeat classification, and Ivanciu et al. (2021) [17], which applied them to ECG-based authentication, this study adapts the Siamese network architecture to address the dual objective: capturing general patterns shared across time-series data while preserving unique, individual-specific characteristics. This approach bridges the gap between modeling population-level trends and subject-specific variability. The Siamese network, with its inherent ability to compare paired inputs, provides an ideal framework to investigate this balance between shared and unique features in ECG signals.

3.3.2 Model Architecture

The Siamese network used in this study consists of two identical branches with shared weights, ensuring that both inputs are processed uniformly. Each branch processes a pair of ECG signals, either positive pairs (signals from the same subject) or negative pairs (signals from different subjects), and extracts embeddings that represent the signals in a high-dimensional space.

Each branch is composed of convolutional layers with residual connections and Squeeze-and-Excitation (SE) blocks, which enhance the networks ability to learn meaningful representations of the input signals. Residual connections, initially introduced in ResNet architectures, allow the network to bypass certain layers by adding shortcut connections, which mitigate the vanishing gradient problem in deep networks. This design ensures efficient gradient flow during backpropagation and facilitates feature learning, as demonstrated in related applications [37, 41].

Squeeze-and-Excitation (SE) blocks enhance a network's feature extraction capabilities by adaptively recalibrating channel-wise feature responses [15]. These blocks perform a two-step mechanism: the "squeeze" operation computes global feature representations by averaging spatial information across channels, and the "excitation" operation applies channel-wise weighting to emphasize the most critical features. In the context of ECG analysis, SE blocks improve classification performance by focusing on the most relevant signal components [11, 34].

The outputs of the two branches of the Siamese network are embeddings, denoted as (Z_1, Z_2) which correspond to the two input signals. These embeddings represent the input signals in a high-dimensional feature space. The similarity between the two embeddings is quantified using the Euclidean distance metric, a simple yet effective measure for assessing pairwise differences in the embedding space.

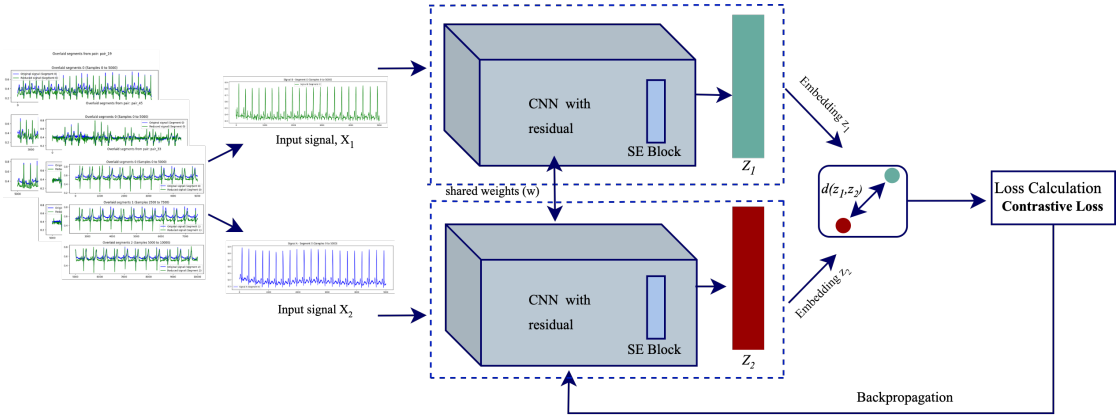


Figure 3.3: Overview of the Siamese network flow and training process. Input signals (X_1, X_2) are paired and processed through identical CNN branches with residual connections and SE blocks, producing embeddings (Z_1, Z_2). The embeddings are compared using the Euclidean distance metric ($d(Z_1, Z_2)$), and the similarity score determines whether the pair is classified as positive or negative. The network is trained using the contrastive loss function, which minimizes the distance for positive pairs and maximizes it for negative pairs. The calculated loss is propagated back through the network during backpropagation to update the weights, ensuring improved performance in distinguishing between signals.

3.3.3 Predictions and loss calculation

It is expected that positive pairs (signals from the same subject) will have embeddings closer together, while negative pairs (signals from different subjects) will be pushed apart in the embedding space. This separation is guided and optimized using the contrastive loss function:

$$\mathcal{L}(y, d) = y \cdot d^2 + (1 - y) \cdot \max(0, m - d)^2 \tag{3.1}$$

Here, y represents the pair label (1 for positive, 0 for negative), and d is the Euclidean distance between embeddings. The margin m enforces separation between negative pairs, ensuring the network learns to differentiate effectively.

The contrastive loss is backpropagated through the network to adjust the weights iteratively, optimizing the network's ability to distinguish between positive and negative pairs. This optimization process enables the model to capture the subtle differences between general patterns and individual-specific features in ECG signals. The actual effectiveness of this separation is evaluated through experimental results, which assess the network's ability to create well-separated

embeddings for positive and negative pairs.

3.3.4 Training process

The training process involved the preparation of input pairs from the ECG dataset. Positive pairs were constructed using original and reconstructed signals from the same subject. The reconstructed signals aimed to preserve intra-subject characteristics while introducing variability influenced by the reconstruction process. Negative pairs, on the other hand, paired signals from different subjects to encourage the network to capture inter-subject differences.

The paired signals (X_1 and X_2) were passed through the Siamese network branches, producing embeddings (Z_1, Z_2), which were then compared using the Euclidean distance metric (see Fig. 3.8). The contrastive loss function optimized this similarity measure by minimizing the distance for positive pairs and maximizing it for negative pairs.

Training was conducted using the Adam optimizer with a learning rate of 0.001 and a batch size of 32 over 25 epochs. During training, accuracy and loss metrics were monitored to evaluate the model's performance. This iterative process was designed to refine the networks ability to extract robust embeddings, with the expectation that this would lead to effective classification of ECG signals, as demonstrated by the results presented in Chapter 4.

3.4 Adaptation for Abnormality Detection

Abnormality detection in time-series data, particularly in ECG signals, is a critical task with significant implications for real-world applications such as diagnosing arrhythmias. This thesis addresses the challenge of abnormality detection by leveraging embeddings learned through the pre-trained Siamese network, trained using a contrastive learning framework. The primary objective is to validate whether the learned embeddings can effectively differentiate between pairs where abnormalities are present and pairs where they are absent, while balancing global patterns shared across subjects and individual-specific features unique to each subject.

To adapt the network explained in 3.3 for the downstream task of abnormality detection, the pre-trained model was used as a feature extractor. The learned weights were frozen to preserve the embeddings' ability to encode both

global and individual-specific features. Pairs of signals were passed through the network to generate embeddings, which were subsequently processed by a custom classification head. This classification head, optimized using Binary Cross-Entropy (BCE) loss, classified the pairs into two categories: pairs without abnormalities (normal + normal) and pairs with abnormalities present (normal + abnormal). This adaptation reframes abnormality detection as a supervised binary classification task.

Importantly, the adaptation process did not explicitly provide individual signal labels (i.e., whether a signal in the pair was normal or abnormal) to the model during the feature extraction stage. Instead, only the pair-level labels-whether abnormalities were present or absent-were used for training the classification head. This approach ensured that the classification head relied solely on the embeddings generated by the pre-trained Siamese network for decision-making.

The rationale for this approach lies in its ability to validate the central research question of this thesis: whether embeddings learned through contrastive learning can robustly capture intra-subject consistency and inter-subject variability. While contrastive loss was not applied in this stage, the embeddings learned during the contrastive learning phase served as the foundation for this classification task. By evaluating the model's performance using metrics such as accuracy, precision, recall, and AUC, the suitability of the learned embeddings for abnormality detection could be systematically assessed.

Further details on the architectural adaptations, classification head design, and methodology for this task are provided in 3.3.

3.4.1 Custom dataset creation

Following the training of the Siamese network, the MIT-BIH Arrhythmia dataset was further processed to create a custom dataset of individual heartbeats for anomaly detection. A key objective was to construct a balanced dataset by starting with abnormal beats. Among the ECG beats in the dataset, those with a length of 268 samples had the highest distribution of abnormal beats (see Fig. 3.4), making this length a natural choice for the fixed input size required by the convolutional neural network. To achieve this, shorter beats were padded with zeros, while longer beats were truncated, ensuring minimal distortion to the signal's structure.

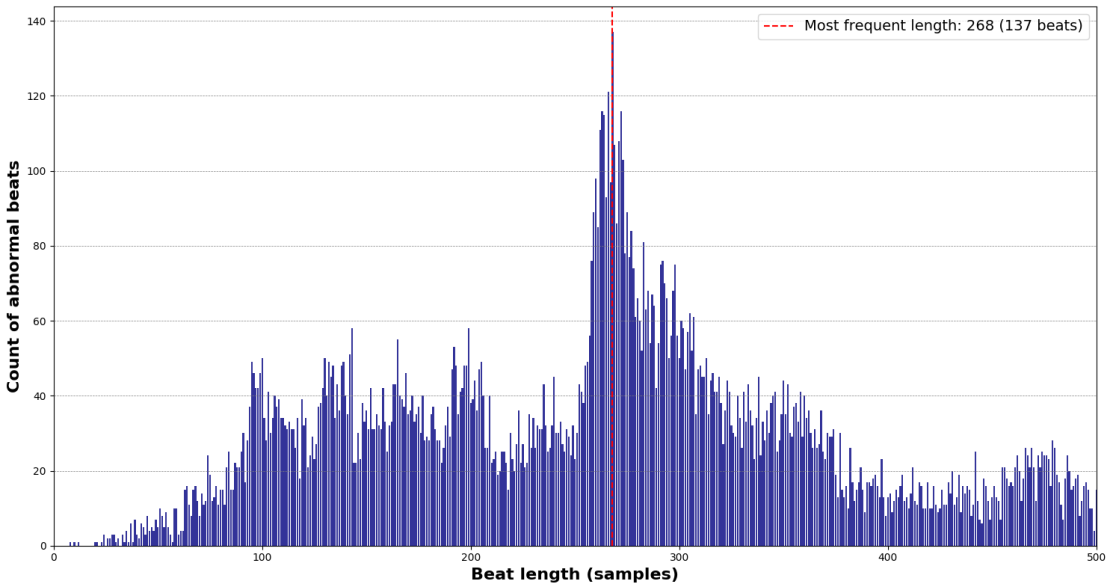


Figure 3.4: Distribution of beat lengths by the count of abnormal beats.

To further increase variability and sample size, beats with lengths close to 268 were included, specifically within the range of 263 to 273 samples. This range provided a balance between preserving subject diversity and maintaining a sufficient number of samples. As shown in Fig. 3.5, the selected range ensured adequate representation of subjects while expanding dataset coverage.

Negative pairs for training were constructed by pairing one abnormal beat with a normal beat. Whenever possible, beats from the same subject were used to ensure the model could learn intra-subject variability. If no normal beats were available for a given subject, a normal beat from a different subject was selected. This pairing strategy was designed to effectively capture both inter-subject and intra-subject variations.

This process resulted in the creation of 1199 positive pairs and 1199 negative pairs, which were used for training the Siamese network.

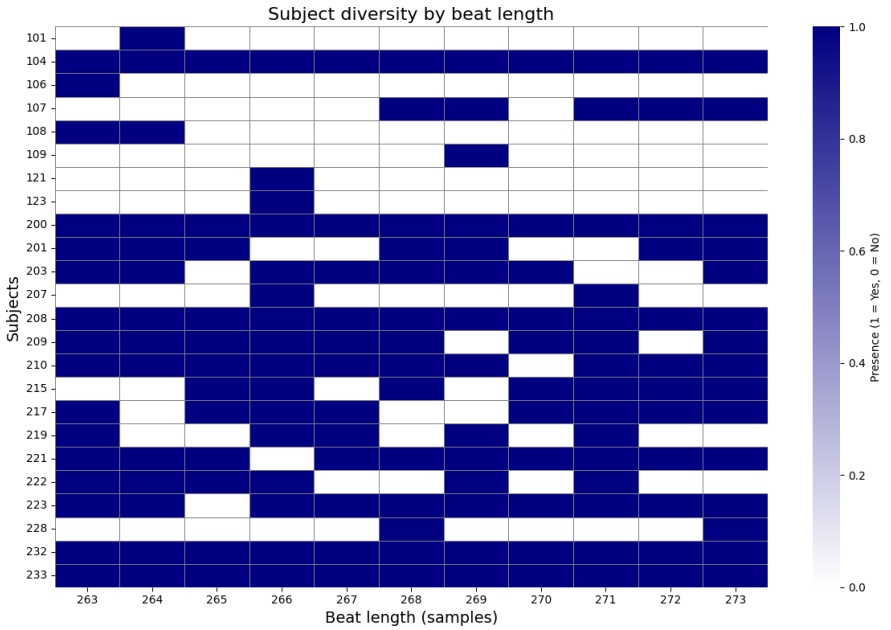


Figure 3.5: Segment length [no. samples] for every subject. This visualization highlights the representation of subjects across the selected range of beat lengths.

3.4.2 Adaptation of the Siamese Network for Abnormality Detection

Following the contrastive learning phase, the pre-trained Siamese network was adapted for the specific task of abnormality detection. The trained model's weights were saved and subsequently used to generate embeddings for pairs of signals:

- **Pairs without abnormalities:** Both signals in the pair are normal (normal + normal).
- **Pairs with abnormalities present:** One signal in the pair is normal, and the other is abnormal (normal + abnormal).

These embeddings, which capture both global patterns and individual-specific features, were then used as input to a custom-defined classification head. The classification head, trained using binary cross-entropy (BCE) loss, classified pairs based on the presence or absence of abnormalities. By leveraging the embeddings learned during the contrastive learning phase, the classification head systematically evaluated the network's ability to distinguish between pairs with intra-subject consistency (pairs without abnormalities) and pairs with inter-subject variability (pairs with abnormalities present).

By reframing abnormality detection as a supervised binary classification task, this adaptation validated the robustness and generalizability of the embeddings for downstream applications. This process highlights how the proposed framework effectively balances generalization across subjects with the preservation of subject-specific details, supporting its applicability to a range of time-series tasks.

Classification head

To adapt the Siamese network for anomaly detection, a classification head was introduced. Unlike projection heads commonly used in contrastive learning [6], which are designed for optimizing contrastive loss, the classification head in this work maps embeddings to binary classification scores optimized using binary cross-entropy (BCE) loss. Its structure is designed to ensure effective learning while mitigating overfitting:

- **Fully Connected Layers:** Dense layers with Swish activation to model complex non-linear relationships within the embedding space.
- **Residual Connections:** These connections improve feature propagation and reduce the risk of vanishing gradients, enhancing training stability.
- **Dropout Regularization:** Dropout layers are applied to prevent overfitting by randomly deactivating neurons during training.
- **Sigmoid Output Layer:** A final layer with sigmoid activation produces a probability score, enabling binary classification of the input signals.

To further adapt the network, the last five layers of the Siamese network were unfrozen for fine-tuning. This allowed selective updates to the pre-trained features, ensuring they were optimized for the anomaly detection task while preserving the robustness of the embeddings learned during contrastive learning.

Training procedure

The combined model, consisting of the partially frozen Siamese network and the classification head, was trained using a binary cross-entropy loss function. To optimize the training process, the following strategies were employed:

- **Learning Rate Scheduler:** An exponential decay learning rate scheduler was used, starting with an initial learning rate of 3×10^{-5}.

- **Early Stopping:** Training was monitored with early stopping to prevent overfitting after 5 epochs without improvement in validation loss.
- **Evaluation Metrics:** In addition to accuracy, evaluation metrics included the Area Under the Receiver Operating Characteristic Curve (AUC), precision, and recall.

3.5 Implementation

3.5.1 Overview of the implementation

The implementation of the Siamese network and the associated preprocessing pipeline was carried out using Python, leveraging its rich ecosystem of libraries for deep learning, signal processing, and data analysis. TensorFlow was chosen as the primary framework for designing and training the network due to its flexibility and GPU acceleration capabilities, which made the training process efficient.

A variety of tools and libraries were utilized during the implementation:

- **TensorFlow:** Facilitated the design, training, and optimization of the Siamese network, enabling seamless GPU acceleration.
- **NumPy and Pandas:** Used for numerical computations and managing ECG data during preprocessing.
- **Scikit-learn:** Assisted with normalization and splitting the dataset into training and testing sets.
- **WFDB:** Provided functions for loading and processing ECG signals from the MIT-BIH Arrhythmia dataset.
- **STL Decomposition:** Used for decomposing ECG signals into their trend, seasonal, and residual components.
- **Visualization Tools:** Libraries such as Matplotlib, Seaborn, and Plotly were employed to visualize ECG signals, decomposition results, and model performance metrics.

The main goal of the implementation was to preprocess ECG signals, train a Siamese network for feature extraction, and evaluate its performance in differentiating between normal and abnormal signals.

3.5.2 Data preprocessing

To prepare the ECG data for training and evaluation, a comprehensive preprocessing pipeline was implemented. The preprocessing steps were designed to normalize the raw ECG signals, decompose them into meaningful components, and save the results for downstream analysis and model training.

The preprocessing involved the following key steps:

1. Signal normalization: To ensure consistency in the data, each ECG signal was normalized using Min-Max normalization. This technique scales the signal to a range between 0 and 1, preserving the signal's original shape while making it suitable for further analysis:

$$x_{normalized} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.2}$$

Subject-by-subject normalization was employed as a potential approach to preserve within-subject characteristics, ensuring that the embeddings capture individual-specific features, which aligns with the research question of distinguishing general patterns from individual fingerprints in time-series data.

2. STL Decomposition: The STL decomposition was implemented using the `statsmodels` library. Below is a snippet of the code used for applying STL decomposition to an ECG signal:

```
from statsmodels.tsa.seasonal import STL

# Function to perform STL decomposition
def stl_decomposition(signal, period=300, seasonal=13):
    stl = STL(signal, period=period, seasonal=seasonal)
    result = stl.fit()
    return result.trend, result.seasonal, result.resid
```

Code 3.1: Function for STL decomposition of a signal

The `STL` object takes the normalized ECG signal as input, along with parameters for the period and seasonal smoothing window. The `fit()` method decomposes the signal into trend, seasonal, and residual components, which are then used in further analysis.

The choice of parameters for STL decomposition was guided by the intrinsic properties of ECG signals and the preprocessing goals for this study. The `period` parameter, set to 300, reflects the approximate periodicity of ECG signals, which corresponds to the heart rate and is measured in terms of samples per cycle. With

the ECG signals sampled at 360 Hz (as per the MIT-BIH Arrhythmia Database), a typical heart rate of 6080 beats per minute results in approximately 300360 samples per cycle. Setting the period close to 300 ensures that the decomposition aligns with the physiological periodicity of the ECG signal.

The `seasonal` parameter, set to 13, controls the window size for capturing short-term fluctuations. This value was chosen empirically to balance sensitivity to subtle variations and the need to smooth noise. A larger value for `seasonal` might oversmooth the data, obscuring important signal details, while a smaller value could overfit to noise. Through exploratory analysis, a value of 13 was found to effectively capture the repeating short-term patterns within the signal without introducing artifacts. Results from varying the `seasonal` parameter and their corresponding visualizations are detailed in Chapter 4.

3.5.3 Signal reconstruction using simplified 1D CNN

The reconstructed ECG signal was obtained by combining the decomposed trend, seasonal, and residual components using a custom 1D convolutional neural network (CNN). The CNN was specifically designed to handle these components in separate branches, emphasizing the unique characteristics of each and combining them to generate a single-channel output signal. This reconstruction step was crucial for reducing noise while retaining meaningful features for downstream tasks, such as anomaly detection.

CNN Architecture:

- **Two branches:** The model employed one branch to process the trend and seasonal components, which share periodic characteristics, and a second branch to handle the residual component, which captures fine-grained irregularities.
- **Residual weighting:** A Lambda layer was used to emphasize the residual component by weighting it more heavily, reflecting its critical role in reconstructing subtle variations in the ECG signal.
- **Final reconstruction:** The outputs from the branches were concatenated and passed through additional convolutional layers to refine the reconstructed signal into a single-channel output.

The modular design of the CNN ensured that the network could efficiently process and combine distinct signal components while minimizing noise. The use of max-pooling layers facilitated feature extraction, while up-sampling layers restored the signal resolution.

Below is a snippet of the Python implementation for the CNN architecture.

```
# Trend and seasonal components branch
trend_seasonal_input = inputs[:, :, :2]
x1 = layers.Conv1D(32, kernel_size=3, activation='relu',
                   padding='same')(trend_seasonal_input)
x1 = layers.BatchNormalization()(x1)
x1 = layers.MaxPooling1D(pool_size=2)(x1)

# Residual component branch
residual_input = inputs[:, :, 2:]
x2 = layers.Conv1D(64, kernel_size=3, activation='relu',
                   padding='same')(residual_input)
x2 = layers.BatchNormalization()(x2)
x2 = layers.MaxPooling1D(pool_size=2)(x2)

# Weight residual component
weighted_residual = layers.Lambda(lambda x: x * 1.5)(x2)

# Combine trend + seasonal and residual components
combined = layers.Concatenate()([x1, weighted_residual])
```

Code 3.2: Multi-branch architecture combining trend, seasonal, and residual components

Training process

- Input signals were divided into overlapping windows of 5000 samples, with a 50% overlap (2500 samples). Overlapping ensured that continuity between reconstructed segments was preserved and avoided sudden transitions at window boundaries.
- The CNN processed each window independently to reconstruct that segment of the signal. These reconstructed segments were then merged using a weighted averaging technique to ensure smoothness in the final reconstructed output.

The overall architectural design of the CNN is shown in Figure 3.6. The model separates the input into branches for processing trend-seasonal components and residual components. The outputs from these branches are combined to produce a single-channel reconstructed signal, which represents the reconstructed version of the input. Additionally, Figure 3.7 provides a 3D visualization of the data flow through the residual branch of the CNN. This visualization highlights how the residual component of the input signal is processed through convolutional layers, batch normalization, pooling layers, and weighting operations, showcasing the branch's role in emphasizing fine-grained signal features.

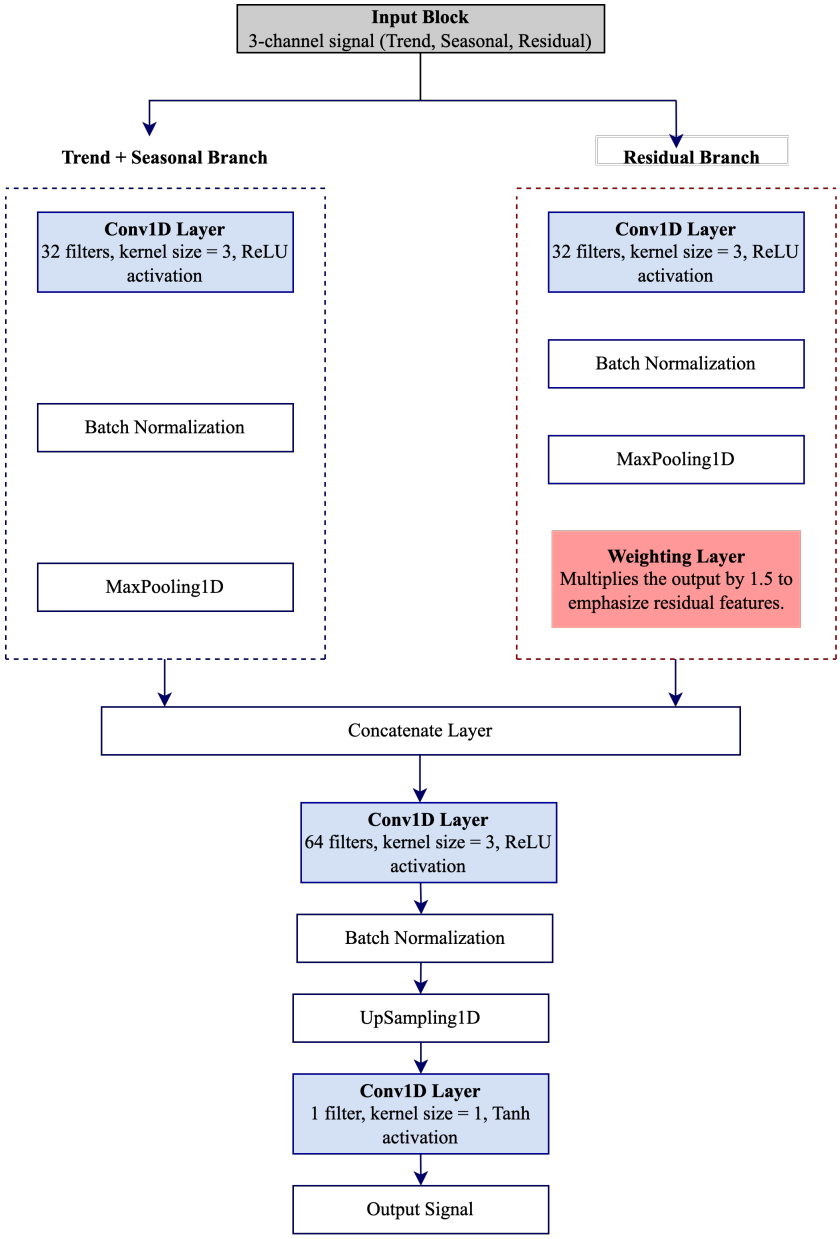


Figure 3.6: Architecture of the simplified 1D CNN for signal reconstruction. The model consists of two branches: one for the trend and seasonal components and another for the residual component, followed by concatenation and further convolutional layers.

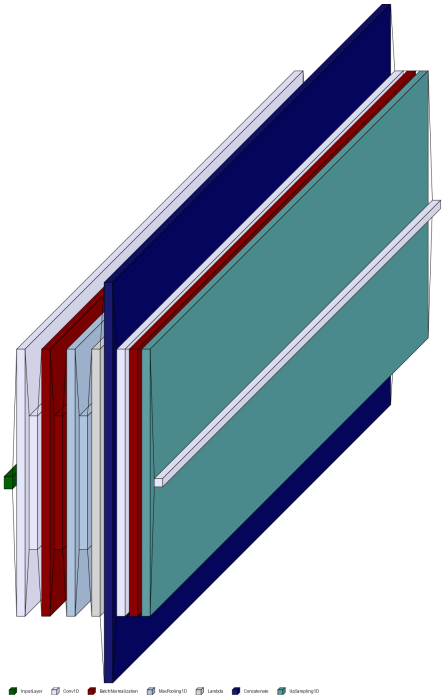


Figure 3.7: 3D visualization of the residual branch in the 1D CNN architecture for processing ECG signals. This branch processes the residual component using a Conv1D layer for feature extraction, BatchNormalization for stability, and MaxPooling1D for downsampling. A Lambda layer scales the residual features by 1.5, amplifying their importance before combining with the trend-seasonal branch.

3.5.4 Implementation of the siamese network

The Siamese network architecture consisted of two identical branches, each designed to process one signal from a pair. These branches employed convolutional layers with residual connections and Squeeze-and-Excitation (SE) blocks to enhance feature extraction.

The network comprised a total of **80 layers**, with the following parameter details:

- **Total parameters:** 1,032,889 (3.94 MB)
- **Trainable parameters:** 1,029,305 (3.93 MB)
- **Non-trainable parameters:** 3,584 (14.00 KB)

The SE block recalibrates channel features using global average pooling and dense layers, enhancing the network's ability to focus on the most informative signal components. Below is the Python implementation of the SE block:

```
def squeeze_excite_block(input_tensor, ratio=16):
    channels = input_tensor.shape[-1]
    x = layers.GlobalAveragePooling1D()(input_tensor)
    x = layers.Dense(channels // ratio, activation='relu')(x)
    x = layers.Dense(channels, activation='sigmoid')(x)
    x = layers.Multiply()([input_tensor, x])
    return x
```

Code 3.3: Squeeze-and-Excitation (SE) block implementation

The residual blocks, a core part of the architecture, consist of two Conv1D layers, each followed by batch normalization. The SE block recalibrates the channel-wise features within the block. A skip connection adds the input tensor directly to the processed output, followed by a ReLU activation. These blocks ensure efficient feature extraction while maintaining gradient flow.

The overall architecture of the Siamese Neural Network is depicted in Figure 3.8. Each branch processes the input through multiple layers, including Conv1D, MaxPooling, and three Residual Blocks with SE blocks. The outputs from the branches are compared using an L2 Distance metric, followed by a Dense layer with a sigmoid activation for binary classification.

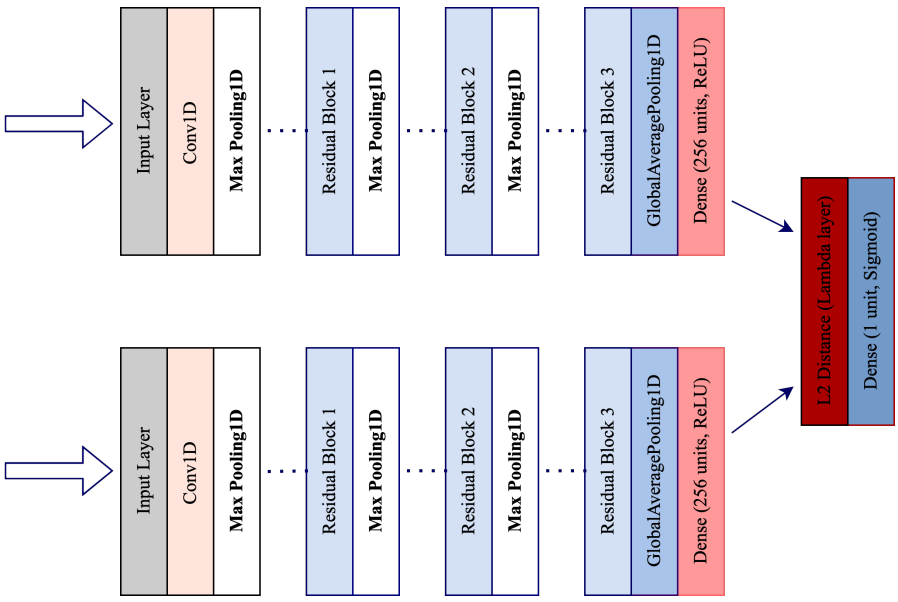


Figure 3.8: Architecture of the Siamese Neural Network.

A detailed view of the first residual block, which plays a crucial role in the network, is shown in Figure 3.9.

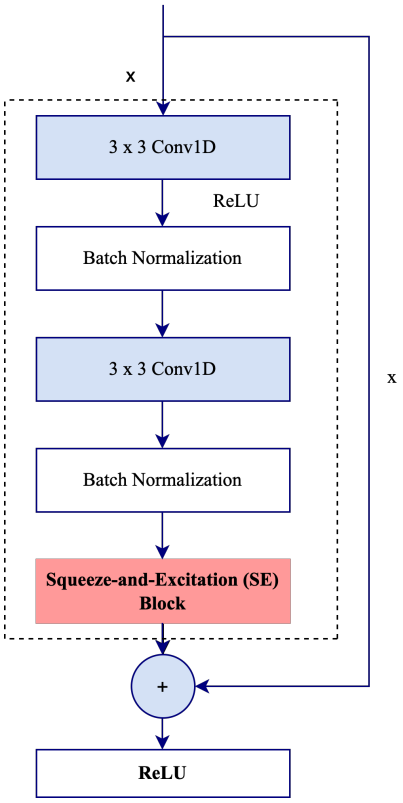


Figure 3.9: Architecture of the first residual block.

To train the network, a **contrastive loss function** was employed. This loss encouraged the network to minimize the distance between embeddings of positive pairs while maximizing the distance for negative pairs. Below is the implementation of the contrastive loss function:

```
def contrastive_loss(margin=1.0):
    def loss(y_true, y_pred):
        square_pred = tf.square(y_pred)
        margin_square = tf.square(tf.maximum(margin - y_pred, 0))
        return tf.reduce_mean(y_true * square_pred + (1 - y_true) *
   margin_square)
    return loss
```

Code 3.4: Implementation of Contrastive Loss

The margin $m = 1.5$ was chosen after initial experimentation and monitoring of the contrastive loss. Despite observed mean distances for positive and negative pairs being smaller than the margin, it provided sufficient flexibility for optimizing separation between embeddings. Further evaluation of embedding separability is detailed in Chapter 4.

Model compilation and training

The network was compiled using the Adam optimizer with a learning rate of 1×10^{-4}. The model was evaluated using the loss metric. The training process involved organizing the dataset into pairs of positive and negative samples and optimizing the contrastive loss function.

During training, the embeddings generated by each branch were compared using the L2 distance, and the resulting similarity scores were used to compute the contrastive loss. The model weights were iteratively updated through backpropagation to minimize the loss.

3.5.5 Classification head and fine-tuning procedure

The adaptation process was implemented as follows:

Embedding Extraction

The embedding layer of the Siamese network was selected as the feature extractor. The weights of the Siamese network were partially frozen to preserve the pre-trained representations, while the last five layers were unfrozen to allow fine-tuning.

Design of the classification head

The classification head was designed with the following structure:

1. A dense layer with 128 neurons, Swish activation, and L2 regularization, followed by Batch Normalization and Dropout (30% rate).

2. A residual connection between two dense layers with matching dimensions to improve gradient flow and feature reuse.

3. A dense layer with 64 neurons, Swish activation, and Dropout (30% rate).

4. A final dense layer with a sigmoid activation function to produce the classification output.

Training configuration

The model was trained with the following parameters, as detailed in Table 3.1:

Table 3.1: Training Configuration Parameters

Parameter	**Value**
Initial Learning Rate	3×10^{-5}
Learning Rate Decay	10%
Loss Function	Binary Cross-Entropy
Optimizer	Adam
Batch Size	32
Epochs	20
Patience for Early Stopping	5 epochs
Dropout Rate	30%
Metrics	Accuracy, AUC, Precision, Recall

Training procedure

The combined model, consisting of the Siamese network and the classification head, was trained on the custom created dataset of labeled ECG beat signals.

The training process included:

- Monitoring the validation loss to ensure optimal model performance.
- Using early stopping to prevent overfitting by halting training after 5 epochs without improvement.
- Evaluating the model using accuracy, precision, recall, and the Area Under the Receiver Operating Characteristic Curve (AUC).

3.5.6 Model evaluation

t-SNE visualization of the embeddings

To evaluate the quality of the embeddings generated by the Siamese network, t-distributed Stochastic Neighbor Embedding (t-SNE) was utilized. This dimensionality reduction technique is widely used to project high-dimensional

embeddings into lower dimensions, enabling a qualitative analysis of their structure [4]. Following the approach used in [6] by Chen et al., where t-SNE was applied to visualize hidden vectors of images from randomly selected classes in the validation set, I used t-SNE to assess the clustering and separability of embeddings from ECG signals.

The embeddings are generated using the Siamese models embedding layer, which is indexed as the second-to-last layer. Positive pairs, representing original ECG signals with their corresponding reconstructed signals, were selected from six randomly chosen subjects. The extracted embeddings were then passed through t-SNE for dimensionality reduction.

For this study, t-SNE was applied to create three-dimensional representations of the embeddings. The implementation used the `scikit-learn` library, with a perplexity parameter dynamically set based on the number of embeddings to balance local and global data structures.

The resulting t-SNE projections were visualized as scatter plots, where each point represents an embedding and is colored according to its subject. These visualizations allow for a qualitative evaluation of how well the Siamese network encodes individual-specific features while maintaining general patterns. By examining the clustering behavior and separability in the embedding space, the t-SNE analysis provides insights into the network's ability to address the central research question of the thesis.

Evaluation of embedding distances

To evaluate the capability of the Siamese network in distinguishing between positive and negative pairs of ECG signals, the Euclidean distance (L2 norm) was computed between the embeddings of paired signals. This measurement quantifies the separation between embeddings of similar (positive) and dissimilar (negative) pairs, providing insight into the network's discriminative performance.

The pre-trained Siamese network was utilized, and its embedding layer was extracted to generate vector representations of the input signals. The distances were computed for two types of pairs:

- **Positive pairs:** Signals belonging to the same class (e.g., from the same subject) were loaded from the directory containing segmented positive pairs. For each pair (*signal A* and *signal B*), embeddings were generated using the embedding model, and the Euclidean distance was calculated

using:

$$d = \sqrt{\sum_{i=1}^{n} (e_{a,i} - e_{b,i})^2}$$

where e_a and e_b represent the embeddings of *signal A* and *signal B*, respectively.

- **Negative pairs:** Signals from different classes (e.g., originating from different subjects) were similarly processed from the directory containing segmented negative pairs, and their distances were computed using the same formula.

For each subject, the distances were stored for subsequent analysis. Histograms were generated to visualize the distribution of distances for positive and negative pairs, aiding in the interpretation of the network's performance.

The evaluation was further extended to include metrics across all subjects to gain a comprehensive understanding of the model's performance. Positive and negative pairs for each subject were processed to compute metrics such as the mean, median, minimum, and maximum distances.

Performance evaluation metrics

To assess the performance of the Siamese network in distinguishing between positive and negative pairs of ECG signals, several evaluation metrics were employed. These metrics provide a quantitative understanding of the model's ability to correctly classify pairs based on the computed distances:

- **Accuracy:** The proportion of correctly classified pairs (positive and negative) over the total number of pairs evaluated.
- **Precision:** The proportion of correctly classified positive pairs among all pairs classified as positive, measuring the network's ability to avoid false positives.
- **Recall (Sensitivity):** The proportion of correctly classified positive pairs among all actual positive pairs, evaluating the model's ability to identify true positives.
- **F1-Score:** The harmonic mean of precision and recall, offering a balanced metric for imbalanced datasets.
- **ROC-AUC (Receiver Operating Characteristic - Area Under Curve):** A measure of the model's ability to distinguish between positive and negative pairs across varying thresholds.

4

Results

The results presented in this chapter are a direct reflection of the methodologies outlined in Chapter 3. Specifically, the evaluation of embeddings generated by the Siamese network, as described in Section 3.5.6, serves as a foundation for understanding the model's ability to distinguish between general patterns and individual fingerprints in time-series data.

This chapter is organized as follows:

- **Signal augmentation and pair creation:** The use of STL decomposition for signal preprocessing, including seasonal parameter tuning, is discussed, highlighting its impact on creating high-quality positive and negative pairs.
- **Visualization of positive and negative pairs:** An assessment is provided to showcase intra-subject consistency and inter-subject variability.
- **Visualization of embedding distributions:** The t-SNE visualizations illustrate the separability of embeddings for positive and negative pairs, offering insights into the network's learned representation.
- **L2 Distance distributions and signal waveforms:** The detailed analysis of L2 distance distributions, supported by signal waveform comparisons.
- **Contrastive Loss:** The loss curve is presented to evaluate the network's convergence during training, demonstrating its effectiveness in learning meaningful representations.
- **Adaptation for anomaly detection: Performance evaluation:** The Siamese network's adaptation for anomaly detection is evaluated using metrics such as accuracy, precision, recall, and AUC.

4.1 Signal augmentation and pair creation

4.1.1 STL Decomposition and seasonal parameter tuning

Figures 4.1, 4.2, and 4.3 illustrate the results of STL decomposition for a representative ECG signal segment (111.dat) at different values of the `seasonal` parameter (`seasonal` = 5, 13, 27).

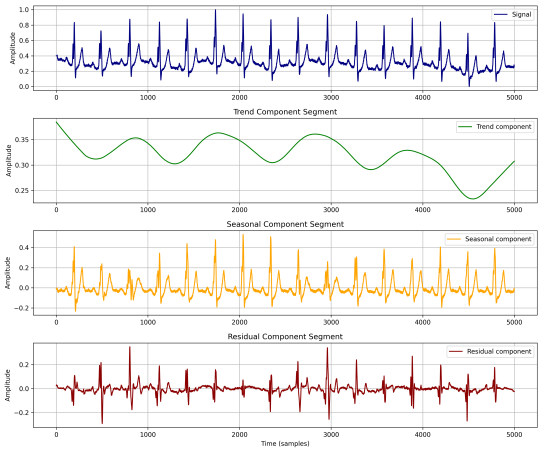


Figure 4.1: STL decomposition with `seasonal` = 5.

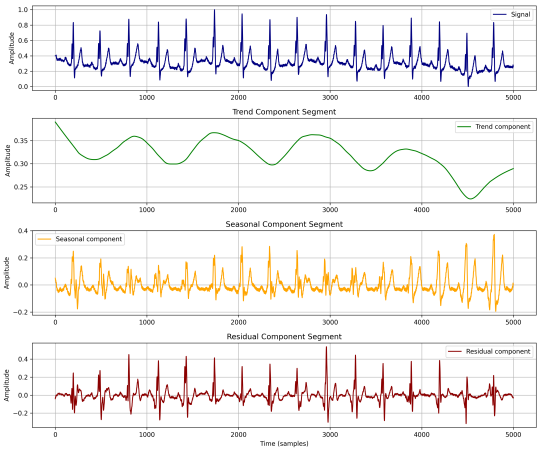


Figure 4.2: STL decomposition with `seasonal` = 13.

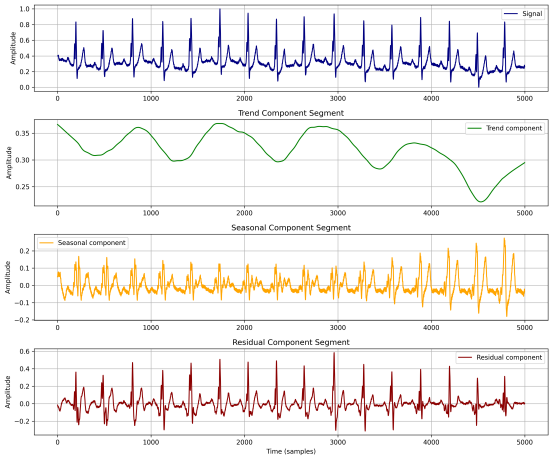


Figure 4.3: STL decomposition with `seasonal` = 27.

These visualizations highlight the effects of parameter selection on the decomposition:

- **`seasonal` = 5**: Captures rapid fluctuations but introduces irregularities into the residual component, indicating potential overfitting to noise.(Figure 4.1).

- **`seasonal = 13`**: Strikes a balance between isolating periodic patterns (e.g., QRS complexes) and maintaining smooth residuals. This value was chosen for downstream tasks as it effectively preserved the signals periodic features (Figure 4.2).

- **`seasonal = 27`**: With this value, the seasonal component smooths the signal too much, leading to an oversimplification of the periodic patterns. As a result, important details that could help differentiate between general patterns and individual-specific features could be lost(Figure 4.3).

Given this, the choice of `seasonal = 13` was guided by exploratory analysis and its alignment with the research question. As shown in Figure 4.2, this value effectively balances capturing the periodic patterns of ECG signals while avoiding oversmoothing or retaining unnecessary noise. The trend and residual components remain well-defined, preserving potential subject-specific features essential for addressing the research question of distinguishing individual fingerprints from general patterns.

4.1.2 Visualization of positive and negative pairs

Figures 4.4 and 4.5 present examples of positive and negative pairs utilized during the training phase. These visualizations provide insight into how the model captures intra-subject consistency and inter-subject variability, which are essential for achieving the research objective. The positive pairs illustrate how the 1D CNN-based reconstruction process, described in Section 3.5.3, effectively retains key periodic features of the ECG signals, ensuring consistency within the same subject. In contrast, the negative pairs emphasize the inherent variability across subjects.

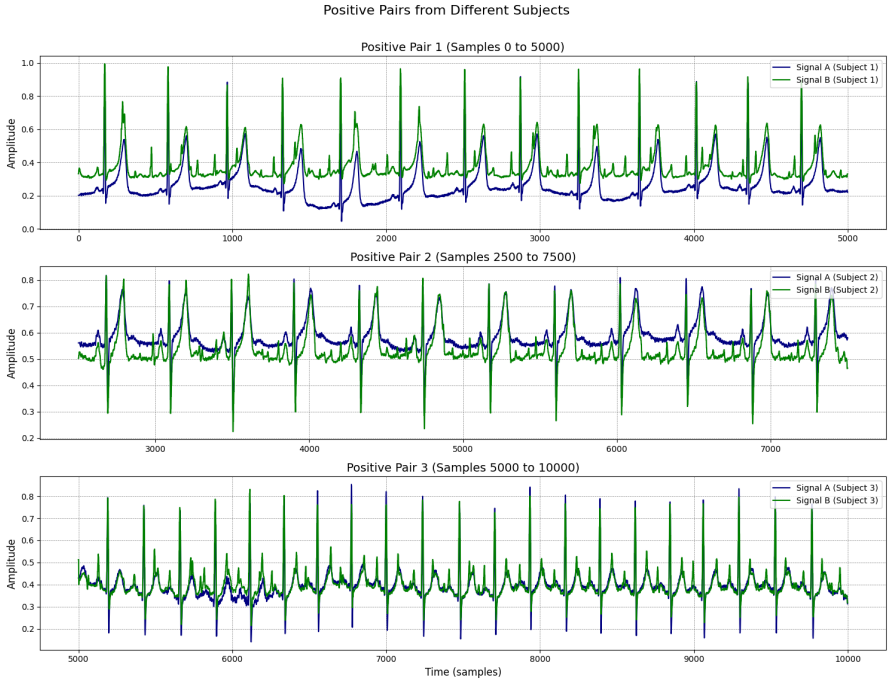


Figure 4.4: Examples of positive pairs: Each pair consists of an original ECG signal and its reconstructed counterpart generated using the 1D CNN described in Section 3.5.3.

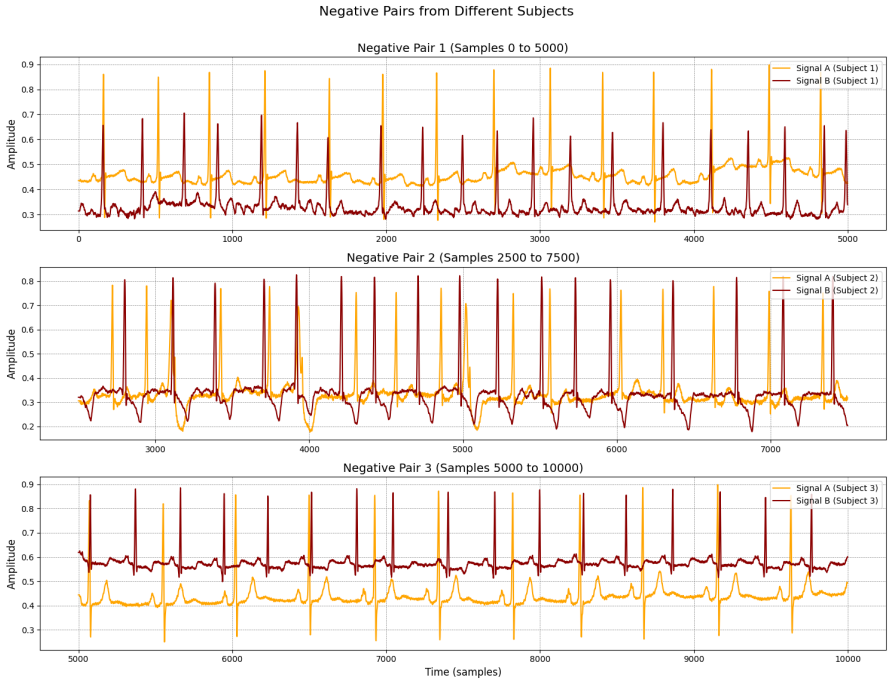


Figure 4.5: Examples of negative pairs: Each pair consists of ECG signals from different subjects.

4.1.3 Visualization of embedding distributions

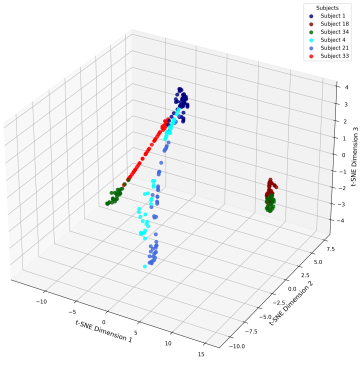
To evaluate the effectiveness of the pair creation process for contrastive learning, 3D t-SNE visualizations were generated to analyze the embeddings produced by the model for different subjects. The goal was to verify if embeddings of positive pairs form compact clusters, while embeddings of the corresponding negative pairs are well-separated.

The t-SNE visualizations in Figure 4.6 display the clustering behavior of embeddings for six subjects. Each subject is represented by a unique color to facilitate comparison. The results suggest that the pair creation process supports the learning of embeddings that capture subject-specific patterns while achieving reasonable separability between different subjects. However, minimal overlaps were observed, indicating that further refinements, such as improving the embedding dimensionality or optimizing the loss function, may enhance performance.

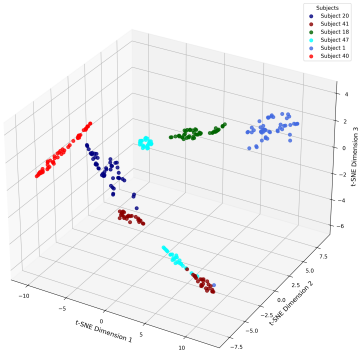
The t-SNE visualizations, presented in Figure 4.6, illustrate the learned embeddings for multiple subjects. For most subjects, embeddings of positive pairs formed relatively cohesive clusters, indicating that the embeddings capture individual fingerprints effectively. This highlights the models ability to emphasize intra-subject similarities while suppressing shared patterns.

Clusters corresponding to different subjects were generally well-separated, showcasing the model's capacity to distinguish between individual fingerprints. However, overlaps were observed in some cases, as seen in Figures 4.6a and 4.6d. These overlaps may stem from segments with similar general patterns or less pronounced individual features. While the degree of overlap is relatively not huge, it suggests that the embedding process could benefit from further optimization or additional preprocessing steps to enhance separability.

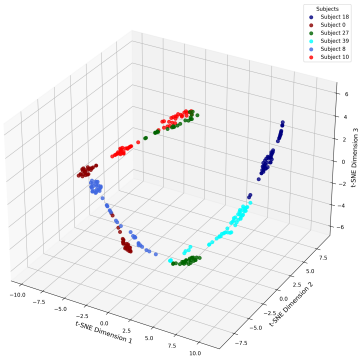
Overall, these findings provide promising evidence that the contrastive learning framework, combined with the pair creation process, helps distinguish between general patterns and individual fingerprints in time-series data. Although there is room for improvement, the clustering of positive pairs and the general separation between subjects indicate that the method holds significant potential for addressing the research question.



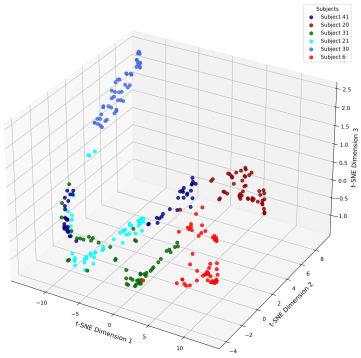
(a) Embeddings of six subjects illustrating clear separation of positive and negative pairs.



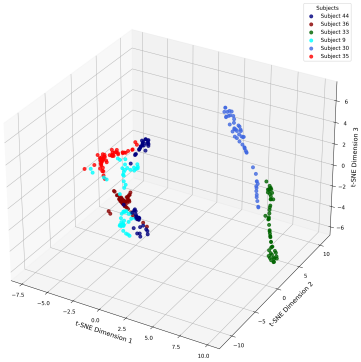
(b) Embedding clusters for another random set of six subjects.



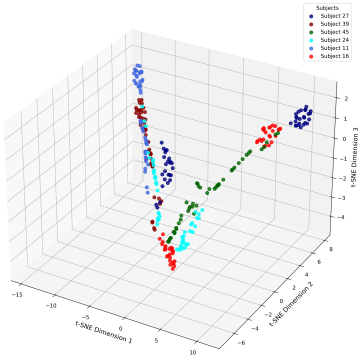
(c) Random subject embeddings showing inter-subject variability.



(d) Another embedding visualization emphasizing intra-subject consistency.



(e) Embedding distributions reflecting variability across pairs.



(f) Final visualization illustrating separation of pair types.

Figure 4.6: 3D t-SNE visualizations of embeddings for various subject groups. Each subplot presents embeddings for a different randomly selected subset of subjects, demonstrating the consistency of embedding separations across multiple random samples.

4.1.4 L2 Distance Distributions and Signal Waveforms

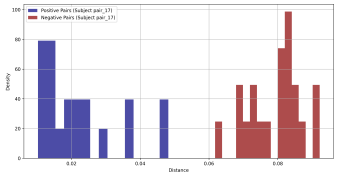
Figures 4.7, 4.8, and 4.9 illustrate the L2 distance histograms and corresponding signal waveforms for three selected subjects. In particular, all negative pairs where the target signal was one of the two signals in the pair were considered, providing an in-depth view of intra-subject and inter-subject variability. These results provide a preliminary overview of the pair creation process and the embeddings generated by the model:

- For **Subject 17**, positive pairs form tightly clustered distances between 0.01 and 0.04, as seen in Figure 4.7(a). The signal waveforms (Figures 4.7(b) and 4.7(c)) confirm high alignment within positive pairs and notable variability in negative pairs. The inclusion of all negative pairs involving Subject 17 highlights the model's ability to capture inter-subject differences.

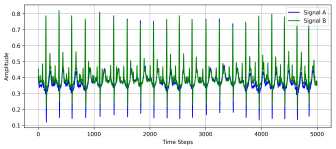
- **Subject 36** demonstrates a distinct clustering of positive distances around 0.04 (Figure 4.8(a)). The waveforms of negative pairs (Figure 4.8(c)) exhibit greater amplitude differences compared to positive pairs.

- For **Subject 21**, a slight overlap is observed in the L2 distance distributions for positive and negative pairs (Figure 4.9(a)). Despite this, the positive pairs maintain relatively consistent waveforms (Figure 4.9(b)), while negative pairs show variability (Figure 4.9(c)).

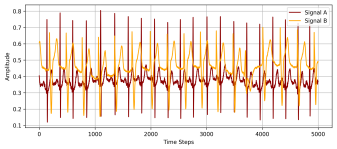
These initial observations highlight the effectiveness of the pair creation process in generating separable embeddings, while also identifying subtle overlaps that suggest opportunities for improvement in embedding refinement.



(a) Histogram of L2 distances for subject 17.

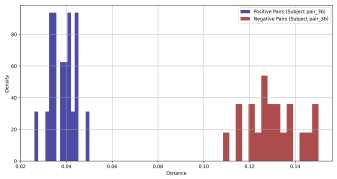


(b) Positive pair for subject 17.

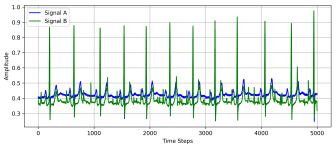


(c) Negative pair for subject 17.

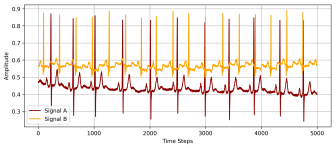
Figure 4.7: Results for Subject 17.



(a) Histogram of L2 distances for subject 36.

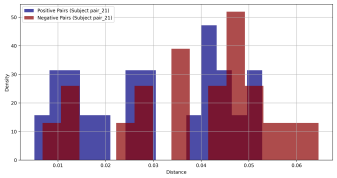


(b) Positive pair for subject 36.

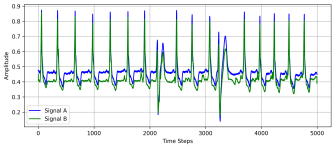


(c) Negative pair for subject 36.

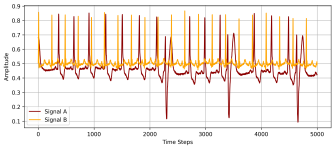
Figure 4.8: Results for Subject 36.



(a) Histogram of L2 distances for subject 21.



(b) Positive pair for subject 21.



(c) Negative pair for subject 21.

Figure 4.9: Results for Subject 21.

Overall metrics summary

Table 4.1 provides a summary of overall metrics, including mean, median, minimum, and maximum distances for positive and negative pairs across all subjects. These metrics highlight the model's ability to generate distinct embeddings while identifying subtle overlaps in certain cases.

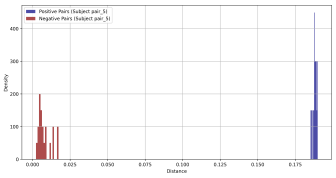
Table 4.1: Overall distance metrics summary for positive and negative pairs.

Metric	Positive	Negative
Mean	0.069459	0.082436
Median	0.069131	0.082067
Min	0.002475	0.002313
Max	0.235245	0.240008

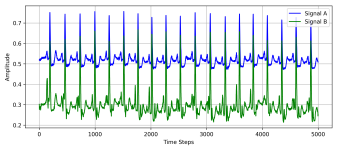
Analysis of meaningful pairs

Figures 4.10, 4.11, 4.12, and 4.13 present the histograms and signal waveforms for pairs selected based on notable metrics. Specifically, these include the positive pair with the highest mean distance (pair 34, mean distance = 0.221959), the positive pair with the smallest mean distance (pair 35, mean distance = 0.009047),

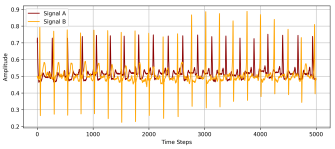
the negative pair with the smallest distance (pair 5, distance = 0.006027), and the negative pair with the largest distance (pair 15, distance = 0.228464). These pairs were chosen to highlight the extremes in distance metrics and provide further insights into the embeddings' behavior and pair creation process.



(a) Histogram of L2 distances for subject 5.

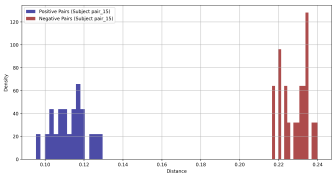


(b) Positive pair for subject 5.

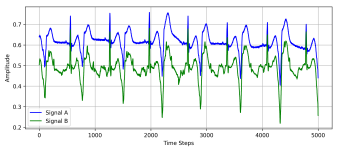


(c) Negative pair with the smallest distance (0.006027).

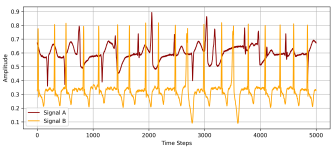
Figure 4.10: Results for Subject 5: Illustrates the negative pair with the smallest distance.



(a) Histogram of L2 distances for subject 15.

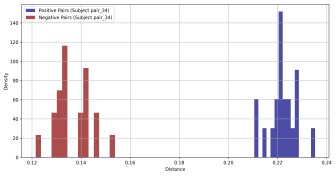


(b) Positive pair for subject 15.

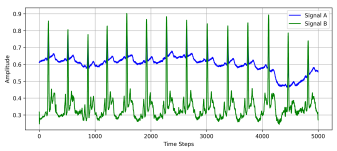


(c) Negative pair with the largest distance (0.228464).

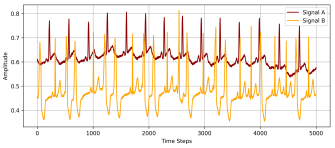
Figure 4.11: Results for Subject 15: Illustrates the negative pair with the largest distance.



(a) Histogram of L2 distances for subject 34.

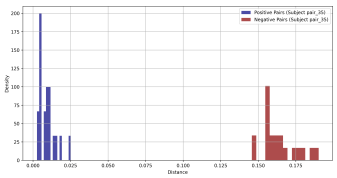


(b) Positive pair with the highest mean distance (0.221959).

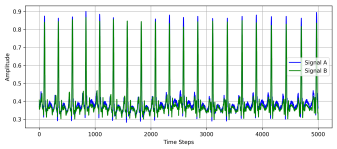


(c) Negative pair for subject 34.

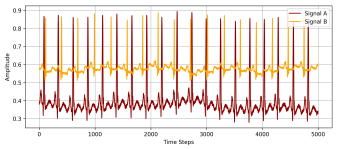
Figure 4.12: Results for Subject 34: Highlights the positive pair with the highest mean distance.



(a) Histogram of L2 distances for subject 35.



(b) Positive pair with the smallest mean distance (0.009047).



(c) Negative pair for subject 35.

Figure 4.13: Results for Subject 35: Highlights the positive pair with the smallest mean distance.

4.1.5 Contrastive Loss

The training loss of the Siamese network, illustrated in Figure 4.14, corresponds to the contrastive loss function, which measures the similarity and dissimilarity between embeddings of positive and negative pairs, respectively. This loss curve reflects the network's capacity to optimize embeddings for both intra-subject consistency and inter-subject separability.

The sharp decline in the initial epochs indicates that the model quickly learns to minimize the distance between embeddings of positive pairs while maximizing the distance for negative pairs.

As training progresses, the loss decreases more gradually, suggesting that the model approaches convergence. The steady decline highlights the effectiveness of the contrastive learning framework in capturing subject-specific patterns and distinguishing between general characteristics. However, the curve also suggests that further refinements, such as additional regularization or adaptive learning rates, could be beneficial to achieve even better generalization.

The margin $m = 1.5$, as discussed in 3.5.4, was selected based on observation of the contrastive loss over multiple training epochs and an evaluation of the resulting embeddings. Although the mean distances for positive pairs (0.069) and negative pairs (0.082) were significantly smaller than the margin, the embeddings displayed clear separation. This larger margin was chosen to allow greater flexibility during training, enabling the network to separate negative pairs without overly constraining their distances. t-SNE visualization further validated this choice, showing tight clusters for positive pairs and well-separated negative pairs. Moreover, the steady decline in the contrastive loss throughout training demonstrated effective optimization under this margin setting.

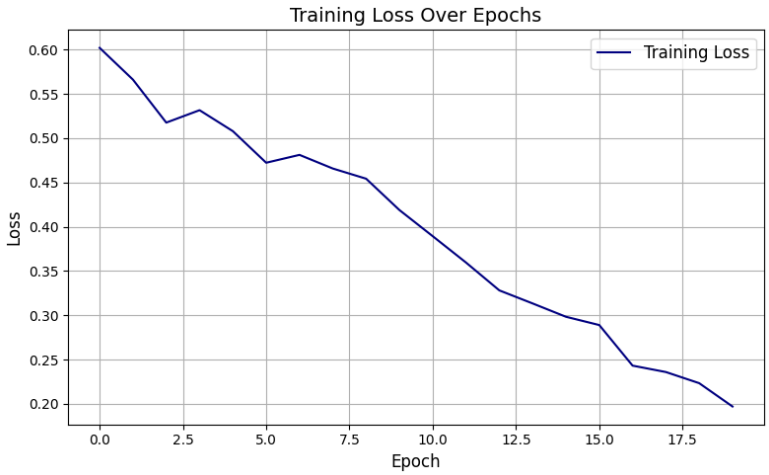


Figure 4.14: Contrastive loss of the Siamese network during training.

4.1.6 Adaptation for anomaly detection: performance evaluation

To evaluate the performance of the adapted Siamese network for anomaly detection, the classification head was fine-tuned using the positive and negative pairs of the custom dataset explained in section 3.4.1. The performance of the model was analyzed using metrics such as accuracy, recall, precision, area under the curve (AUC), and loss over the training epochs.

Training and validation loss

Figure 4.15 shows the training and validation loss curves over the epochs. The loss steadily decreases, reflecting the model's ability to minimize the error between the predicted and actual classifications. A notable observation is the divergence between training and validation loss starting around epoch 10, indicating the onset of potential overfitting. This divergence aligns with the trends observed in the AUC curves (Figure 4.18), where the validation AUC reached near-perfect levels around epoch 15.

To address this, regularization techniques, including dropout and learning rate scheduling, were applied to mitigate overfitting. The training was stopped at 20 epochs based on the validation AUC reaching saturation, ensuring a balance between performance and generalization while preventing the model from overfitting to the training data.

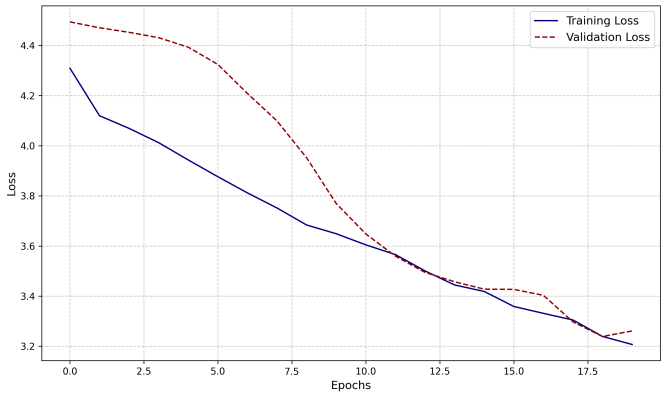


Figure 4.15: Training and Validation Loss over Epochs.

Recall performance

The recall curves for training and validation data, shown in Figure 4.16, demonstrate the model's ability to identify true positive cases effectively. While recall is high across most epochs, the slight decline in validation recall towards the end suggests room for further optimization of the model's generalization capability.

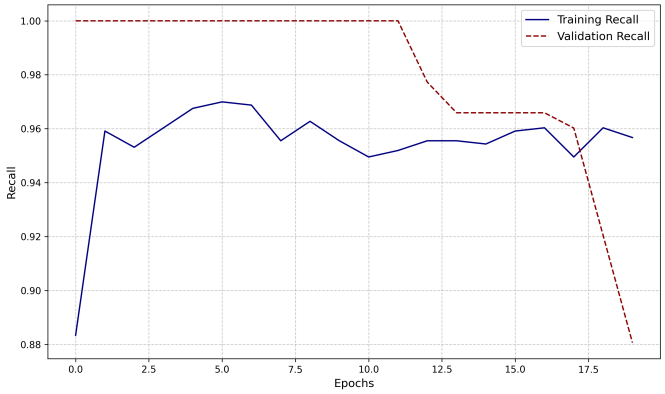


Figure 4.16: Training and Validation Recall over Epochs.

Precision performance

Figure 4.17 illustrates the precision curves for training and validation datasets. Precision improves steadily during the training process, with the validation precision closely following the training curve. The model shows promising performance in maintaining a high precision, particularly in later epochs.

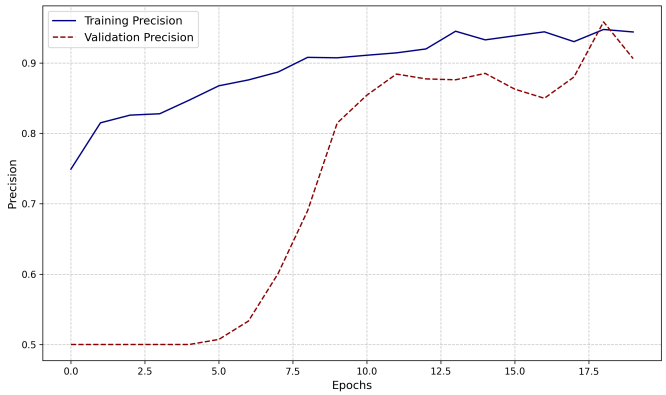


Figure 4.17: Training and Validation Precision over Epochs.

Area Under the Curve (AUC)

The AUC curves, depicted in Figure 4.18, provide a comprehensive evaluation of the model's discriminative ability across varying thresholds. The consistent increase in AUC for both training and validation datasets demonstrates the effectiveness of the model in distinguishing between anomalous and non-anomalous signals.

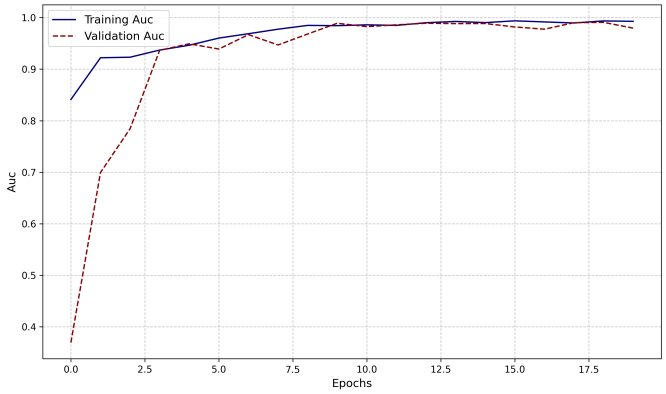


Figure 4.18: Training and Validation AUC over Epochs.

Accuracy Performance

The training and validation accuracy curves are shown in Figure 4.19. The model achieves high accuracy across both datasets, with the validation accuracy stabilizing after the initial epochs. This stability indicates that the model effectively generalizes to unseen data.

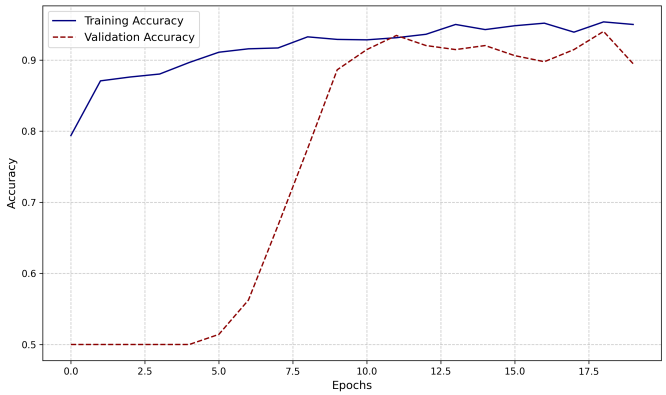


Figure 4.19: Training and Validation Accuracy over Epochs.

The performance evaluation demonstrates that the adapted Siamese network with a classification head shows promising results in anomaly detection. High recall and precision indicate that the model effectively minimizes false negatives and false positives. However, minor divergences in validation metrics, such as recall and loss, highlight areas for further improvement, particularly in generalization. Future work could explore more robust regularization techniques and additional fine-tuning strategies to enhance performance further.

4.1.7 Final results of anomaly detection

The final evaluation of the adapted Siamese network was performed on a test set consisting of positive and negative pairs. The performance metrics obtained from the evaluation are summarized below:

- **Accuracy:** 93.65%
- **Precision:** 97.59%
- **Recall:** 89.50%
- **F1 Score:** 93.37%

These results demonstrate that the model performs well in distinguishing between positive and negative pairs. The high precision (>97%) indicates that the model rarely misclassifies negative pairs as positive, minimizing false positives. The recall, although slightly lower (>89%), suggests that the model occasionally misses positive pairs, highlighting an area for potential improvement. The F1 score of 93.37% reflects a strong overall balance between precision and recall.

The performance metrics confirm that the contrastive learning framework, coupled with the classification head, provides a promising approach for anomaly

detection. The high accuracy and precision validate the effectiveness of the model in separating subject-specific patterns from general characteristics. However, the slightly lower recall highlights an opportunity to further enhance the model's sensitivity to positive pairs, possibly through additional fine-tuning or the use of more representative training data.

Overall, these results support the hypothesis that the adaptation of the Siamese network for anomaly detection is effective and align with the promising findings reported during the embedding evaluation phase.

5

Conclusions

The analysis of time-series data, particularly in biomedical applications such as Electrocardiograms (ECG), presents the dual challenge of capturing global patterns shared across subjects while preserving individual-specific features. This thesis set out to address this challenge by exploring the way contrastive learning techniques can help distinguish general patterns from individual fingerprints in time-series data. To this end, a contrastive learning framework tailored to ECG analysis was developed, leveraging advanced preprocessing methods and a deep learning architecture designed to balance global pattern extraction with subject-specific feature retention.

The use of Seasonal-Trend decomposition using Loess (STL) for signal preprocessing demonstrated the importance of parameter tuning in isolating periodic components without compromising critical details. The exploration of seasonal values showcased the ability to retain signal fidelity, ensuring that key ECG features such as the QRS complex were preserved while suppressing noise. This preprocessing step laid the foundation for generating high-quality embeddings.

The Siamese network architecture, enhanced with Residual Blocks and Squeeze-and-Excitation (SE) mechanisms, successfully generated embeddings capable of encoding both intra-subject consistency and inter-subject variability. Through qualitative and quantitative evaluations, including t-SNE visualizations and L2 distance distributions, the embeddings were shown to effectively separate positive and negative pairs. The contrastive loss further validated the network's ability to learn meaningful representations in the embedding space.

Anomaly detection was employed as a representative use case to evaluate the frameworks downstream applicability. By fine-tuning the embeddings with a classification head, the model achieved high accuracy, precision, recall, and AUC, underscoring its potential for real-world applications. However, the focus of this work extends beyond anomaly detection, as the insights gained are applicable to a broader range of time-series tasks requiring fine-grained differentiation.

5.0.1 Contributions and Implications

This thesis makes the following key contributions:

- Development of a contrastive learning framework tailored to time-series data, balancing global pattern extraction with individual-specific feature retention.
- Demonstration of the effectiveness of STL decomposition for ECG signal preprocessing and its role in facilitating downstream tasks.
- Proposal of a Siamese network architecture with Residual and SE Blocks to generate robust embeddings for distinguishing intra-subject and inter-subject variability.
- Validation of the research question through extensive qualitative and quantitative evaluations, including embedding visualizations and distance metrics.

The findings of this thesis extend the understanding of contrastive learning in time-series analysis, providing a framework that can be adapted to various biomedical and non-biomedical domains. By bridging the gap between global pattern recognition and individual feature preservation, this work contributes to advancing the state of the art in time-series analysis.

5.0.2 Future Work

While this thesis lays a robust foundation, there are several avenues for future research:

- Investigating the impact of alternative distance metrics, such as cosine similarity or L1 distance, on embedding quality and downstream performance.
- Evaluating the use of dynamic time warping (DTW) as a distance metric, which could offer insights into temporal misalignment between signals and provide additional robustness in capturing intra- and inter-subject variations [8].

- Expanding the framework to other biomedical signals, such as EEG or respiratory signals, to evaluate its generalizability.
- Exploring domain adaptation techniques to enhance model performance across datasets with varying characteristics.

  In conclusion, this thesis highlights the potential of contrastive learning techniques to address the dual challenge of extracting general patterns and preserving individual-specific features in time-series data. While the results demonstrate significant progress in ECG analysis, the methodologies and insights presented here pave the way for broader applications and future innovations in time-series research.

References

[1] A. M. Abagaro, H. Barki, G. Ayana, et al. "Automated ECG Signals Analysis for Cardiac Abnormality Detection and Classification". In: *Journal of Electrical Engineering & Technology* 19 (2024), pp. 3355–3371. DOI: 10.1007/s42835-024-01902-y.

[2] Q. An et al. "A Comprehensive Review on Machine Learning in Healthcare Industry: Classification, Restrictions, Opportunities and Challenges". In: *Sensors* 23.9 (2023), p. 4178. DOI: 10.3390/s23094178. URL: https://doi.org/10.3390/s23094178.

[3] Lahcen el Bouny, Mohammed Khalil, and Adib Abdellah. "A Wavelet Denoising and Teager Energy Operator-Based Method for Automatic QRS Complex Detection in ECG Signal". In: *Circuits, Systems, and Signal Processing* 39 (Oct. 2020). DOI: 10.1007/s00034-020-01397-8.

[4] T. Tony Cai and Rong Ma. "Theoretical Foundations of t-SNE for Visualizing High-Dimensional Clustered Data". In: *arXiv preprint arXiv:2105.07505* (2021). Submitted on 16 May 2021 (v1), last revised 1 Nov 2022 (this version, v4).

[5] Hui Chen et al. "CLECG: A Novel Contrastive Learning Framework for Electrocardiogram Arrhythmia Classification". In: *IEEE Signal Processing Letters* 28 (2021), pp. 1993–1997. DOI: 10.1109/LSP.2021.3114119.

[6] Ting Chen et al. "A Simple Framework for Contrastive Learning of Visual Representations". In: *CoRR* abs/2002.05709 (2020). arXiv: 2002.05709. URL: https://arxiv.org/abs/2002.05709.

[7] Joseph Y. Cheng et al. "Subject-Aware Contrastive Learning for Biosignals". In: *CoRR* abs/2007.04871 (2020). arXiv: 2007.04871. URL: https://arxiv.org/abs/2007.04871.

[8] G. Cisotto et al. "Dynamic Time Warping in Time-Series Embedding: Applications to Neuroinformatics". In: *Frontiers in Neuroinformatics* (2024). (accepted).

[9] Robert B. Cleveland et al. "STL: A seasonal-trend decomposition". In: *J. Off. Stat.* 6.1 (1990), pp. 3–73.

[10] Matteo Gadaleta et al. "Prediction of atrial fibrillation from at-home single-lead ECG signals without arrhythmias". In: *npj Digital Medicine* 6 (2023), p. 229. DOI: 10.1038/s41746-023-00966-w. URL: https://doi.org/10.1038/s41746-023-00966-w.

[11] Rongjun Ge et al. "Convolutional squeeze-and-excitation network for ECG arrhythmia detection". In: *Artificial Intelligence in Medicine* 121 (2021), p. 102181. DOI: 10.1016/j.artmed.2021.102181.

[12] Ary L. Goldberger et al. "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals". In: *Circulation* 101.23 (2000). Accessed from https://physionet.org/content/1 e215–e220.

[13] Jinpei Han, Xiao Gu, and Benny Lo. "Semi-Supervised Contrastive Learning for Generalizable Motor Imagery EEG Classification". In: *2021 IEEE 17th International Conference on Wearable and Implantable Body Sensor Networks (BSN)*. 2021, pp. 1–4. DOI: 10.1109/BSN51625.2021.9507038.

[14] Z. Han et al. "A Review of Deep Learning Models for Time Series Prediction". In: *IEEE Sensors Journal* 21.6 (2021), pp. 7833–7848. DOI: 10.1109/JSEN.2019.2923982. URL: https://doi.org/10.1109/JSEN.2019.2923982.

[15] Jie Hu, Li Shen, and Gang Sun. "Squeeze-and-Excitation Networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018, pp. 7132–7141. DOI: 10.1109/CVPR.2018.00745. URL: https://openaccess.thecvf.com/content_cvpr_2018/html/Hu_Squeeze-and-Excitation_Networks_CVPR_2018_paper.html.

[16] Steven A. Israel et al. "ECG to identify individuals". In: *Pattern Recognition* 38.1 (2005), pp. 133–142. DOI: 10.1016/j.patcog.2004.05.014.

[17] L. Ivanciu, IA. Ivanciu, P. Farago, et al. "An ECG-based Authentication System Using Siamese Neural Networks". In: *J. Med. Biol. Eng.* 41 (2021), pp. 558–570. DOI: 10.1007/s40846-021-00637-9.

[18] Ashish Jaiswal et al. "A Survey on Contrastive Self-Supervised Learning". In: *Technologies* 9.1 (2021), p. 2. DOI: 10.3390/technologies9010002. URL: https://www.mdpi.com/2227-7080/9/1/2.

[19] Aofan Jiang et al. *Anomaly Detection in Electrocardiograms: Advancing Clinical Diagnosis Through Self-Supervised Learning*. 2024. arXiv: 2404.04935 [cs.CV]. URL: https://arxiv.org/abs/2404.04935.

[20] Dani Kiyasseh, Tingting Zhu, and David A. Clifton. "CLOCS: Contrastive Learning of Cardiac Signals Across Space, Time, and Patients". In: *Proceedings of the 38th International Conference on Machine Learning*. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 5606–5615. URL: http://proceedings.mlr.press/v139/kiyasseh21a.html.

[21] Prakhar Kumar, Pranjal Rawat, and Siddharth Chauhan. "Contrastive self-supervised learning: review, progress, challenges and future research directions". In: *International Journal of Multimedia Information Retrieval* 11 (2022), pp. 461–488. DOI: 10.1007/s13735-022-00245-6. URL: https://doi.org/10.1007/s13735-022-00245-6.

[22] Yeseul Lee, Yunseon Byun, and Jun-Geol Baek. "Time Series Anomaly Detection Using Contrastive Learning based One-Class Classification". In: *2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. 2023, pp. 330–335. DOI: 10.1109/ICAIIC57133.2023.10067089.

[23] Yeseul Lee, Yunseon Byun, and Jun-Geol Baek. "Time Series Anomaly Detection Using Contrastive Learning based One-Class Classification". In: *2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. 2023, pp. 330–335. DOI: 10.1109/ICAIIC57133.2023.10067089.

[24] H. Li and P. Boulanger. "A Survey of Heart Anomaly Detection Using Ambulatory Electrocardiogram (ECG)". In: *Sensors* 20.5 (2020), p. 1461. DOI: 10.3390/s20051461.

[25] Xiaofeng Liu et al. "Semi-Supervised Contrastive Learning for Time Series Classification in Healthcare". In: *IEEE Transactions on Emerging Topics in Computational Intelligence* PP (May 2024), pp. 1–14. DOI: 10.1109/TETCI.2024.3400885.

[26] Ziyu Liu et al. "Self-supervised contrastive learning for medical time series: A systematic review". In: *Sensors* 23.9 (2023), p. 4221. DOI: 10.3390/s23094221. URL: https://www.mdpi.com/1424-8220/23/9/4221.

[27] M. Mahajan, S. Kadam, V. Kulkarni, et al. "ECG signal classification via ensemble learning: addressing intra and inter-patient variations". In: *International Journal of Information Technology* 16 (2024), pp. 4931–4939. DOI: 10.1007/s41870-024-02086-4.

[28] Sherin M. Mathews, Chandra Kambhamettu, and Kenneth E. Barner. "A novel application of deep learning for single-lead ECG classification". In: *Computers in Biology and Medicine* 99 (2018), pp. 53–62. DOI: 10.1016/j.compbiomed.2018.05.013.

[29] Katie Matton et al. "Contrastive Learning of Electrodermal Activity Representations for Stress Detection". In: *Proceedings of the Conference on Health, Inference, and Learning*. Vol. 209. PMLR. PMLR, 2023, pp. 410–426. URL: https://proceedings.mlr.press/v209/matton23a.html.

[30] G.B. Moody and R.G. Mark. "The impact of the MIT-BIH Arrhythmia Database". In: *IEEE Engineering in Medicine and Biology Magazine* 20.3 (2001), pp. 45–50. DOI: 10.1109/51.932724.

[31] Mohammad Amin Morid, Olivia R. Liu Sheng, and Joseph Dunbar. "Time Series Prediction Using Deep Learning Methods in Healthcare". In: 14.1 (2023). ISSN: 2158-656X. DOI: 10.1145/3531326. URL: https://doi.org/10.1145/3531326.

[32] A.K. Munnangi et al. "Survival study on deep learning techniques for IoT enabled smart healthcare system". In: *Health Technol.* 13 (2023), pp. 215–228. DOI: 10.1007/s12553-023-00736-4. URL: https://doi.org/10.1007/s12553-023-00736-4.

[33] Jungwoo Oh et al. *Lead-agnostic Self-supervised Learning for Local and Global Representations of Electrocardiogram*. 2022. arXiv: 2203.06889 [cs.LG]. URL: https://arxiv.org/abs/2203.06889.

[34] J. Park et al. "ECG-Signal Multi-Classification Model Based on Squeeze-and-Excitation Residual Neural Networks". In: *Applied Sciences* 10.18 (2020), p. 6495. DOI: 10.3390/app10186495.

[35] Ian Peate. *Anatomy and Physiology for Nursing and Healthcare Students at a Glance*. John Wiley & Sons, 2022.

[36] Brent G. Petty. *Basic Electrocardiography*. Springer Nature, 2020.

[37] Wei Qu et al. "A Residual Based Attention Model for EEG Based Sleep Staging". In: *IEEE Journal of Biomedical and Health Informatics* 24.10 (2020), pp. 2833–2843. DOI: 10.1109/JBHI.2020.2978004.

[38] Suha Rabbani and Naimul Khan. "Contrastive Self-Supervised Learning for Stress Detection from ECG Data". In: *Bioengineering* 9.8 (2022), p. 374. DOI: 10.3390/bioengineering9080374. URL: https://doi.org/10.3390/bioengineering9080374.

[39] Xinke Shen et al. "Contrastive Learning of Subject-Invariant EEG Representations for Cross-Subject Emotion Recognition". In: *IEEE Transactions on Affective Computing* 14.3 (2023), pp. 2496–2511. DOI: 10.1109/TAFFC.2022.3164516.

[40] V. F. Silva, M. E. Silva, P. Ribeiro, et al. "Novel features for time series analysis: a complex networks approach". In: *Data Mining and Knowledge Discovery* 36 (2022), pp. 1062–1101. DOI: 10.1007/s10618-022-00826-3.

[41] Nebras Sobahi et al. "Attention-based 3D CNN with residual connections for efficient ECG-based COVID-19 detection". In: *Computers in Biology and Medicine* 143 (2022), p. 105335. DOI: 10.1016/j.compbiomed.2022.105335.

[42] Shreya Srivastava et al. *rECGnition_v1.0: Arrhythmia detection using cardiologist-inspired multi-modal architecture incorporating demographic attributes in ECG*. 2024. arXiv: 2410.18985 [eess.SP]. URL: https://arxiv.org/abs/2410.18985.

[43] Chi Ian Tang et al. "Exploring Contrastive Learning in Human Activity Recognition for Healthcare". In: *CoRR* abs/2011.11542 (2020). arXiv: 2011.11542. URL: https://arxiv.org/abs/2011.11542.

[44] M. M. Eduardo Vasconcellos et al. "Siamese Convolutional Neural Network for Heartbeat Classification Using Limited 12-Lead ECG Datasets". In: *IEEE Access* 11 (2023), pp. 5365–5376. DOI: 10.1109/ACCESS.2023.3236189.

[45] Muhammad Zubair and Changsun Yoon. "Cost-Sensitive Learning for Anomaly Detection in Imbalanced ECG Data Using Convolutional Neural Networks". In: *Sensors* 22.11 (2022), p. 4075. DOI: 10.3390/s22114075. URL: https://doi.org/10.3390/s22114075.

Acknowledgments

Firstly, I would like to express my gratitude to my supervisors, Professor Cisotto and Dr. Zancanaro, for their support, guidance, and unwavering belief in me.

And to my mom and dad, thank you for your unlimited support, for believing in me even when I doubted myself, and for being my rock through everything. I owe you everything.

To my brothers, Diogjen and Kevin, I'm endlessly grateful for everything you've done for me. Your humor- of course questionable as it may be- and your ability to make me laugh, even at my own expense, have been a constant source of light in my life.

To my big family, from my loving grandparents to my little-but-twice-my-height cousins, thank you for always surrounding me with love.

To all my friends, and especially Eni, your support and friendship have brought me so much joy over the years, and it's meant even more to me during this journey. Thank you for standing by me through thick and thin.

And finally, like the sunflower finds its way to the sun - to my love, Ergys - thank you for sharing this journey with me and for making my life so much brighter. Here's to a lifetime of precious memories yet to be made, and to the happiness I am so grateful to share with you.

To all of you - I hope I've made you proud. Everything I have achieved, and everything I will achieve in the future, is for you and because of you.

Declaration

I hereby declare that all contents and analyses presented in this thesis are my original work. However, to enhance the clarity and quality of the text, I employed OpenAIs ChatGPT for text restyling and proofreading. This support was limited to improving the linguistic presentation, and it did not influence the originality or integrity of the technical and scientific content.