UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS no CENTRO DE CIÊNCAS DE SAÚDE
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA EM SAÚDE

Dominik Sandro Perrin

Classifying Atrial Arrhythmias in Electrocardiograms Evaluating Different Deep Learning Techniques to Find a Simple Well-Performing Approach

Florianópolis
2023

Dominik Sandro Perrin

Classifying Atrial Arrhythmias in Electrocardiograms Evaluating Different Deep Learning Techniques to Find a Simple Well-Performing Approach

Dissertação submetida ao Programa de Pós-Graduação em Mestrado Profissional de Informática em Saúde da Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Mestre em Informática em Saúde.

Orientador: Prof. Jefferson Luiz Brum Marques - PhD

Florianópolis

2023

Perrin, Dominik Sandro
Classifying Atrial Arrhythmias in Electrocardiograms Evaluating Different Deep Learning Techniques to Find a Simple Well-Performing Approach / Dominik Sandro Perrin ; orientador, Jefferson Luiz Brum Marques, 2023.
72 p.

Dissertação (mestrado profissional) - Universidade Federal de Santa Catarina, Centro de Ciências da Saúde, Programa de Pós-Graduação em Informática em Saúde, Florianópolis, 2023.

Inclui referências.

1. Informática em Saúde. 2. Electrocardiogram. 3. Automatic Classification. 4. Deep Learning. 5. Atrial Arrhythmias. I. Marques, Jefferson Luiz Brum. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Informática em Saúde. III. Título.

Dominik Sandro Perrin

Título: Classifying Atrial Arrhythmias in Electrocardiograms Evaluating Different Deep Learning Techniques to Find a Simple Well-Performing Approach

O presente trabalho em nível de Mestrado foi avaliado e aprovado, em 14 de Setembro de 2023, pela banca examinadora composta pelos seguintes membros:

Prof. Jefferson Luiz Brum Marques, PHD
Universidade Federal de Santa Catarina

Profa. Sayonara de Fátima Faria Barbosa, Dra.
Universidade Federal de Santa Catarina

Dr. Rafael Sanchotene Silva
Cadence Design Systems

Certificamos que esta é a versão original e final do trabalho de conclusão que foi julgado adequado para obtenção do título de Mestre em Informática em Saúde.

ICP-Edu
Documento assinado digitalmente
Sayonara de Fatima Faria Barbosa
Data: 11/12/2023 19:35:38-0300
CPF: ***.819.497-**
Verifique as assinaturas em https://v.ufsc.br

Profa. Sayonara de Fátima Faria Barbosa, Dra.
Coordenação do Programa de Pós-Graduação

ICP-Edu
Documento assinado digitalmente
Jefferson Luiz Brum Marques
Data: 10/12/2023 11:17:44-0300
CPF: ***.624.060-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Jefferson Luiz Brum Marques, PhD
Orientador

Florianópolis, 2023

I am dedicating this work to my beloved and my family.

ACKNOWLEDGEMENTS

First, I wish to express my gratitude to the program professors whose lectures and the opportunity to participate in their classes made my studies unforgettable. I especially want to thank the course coordinators, Professors Grace and Sayonara, for their guidance and help through all the administrative procedures and for making my exchange possible. Additionally, I would like to express my appreciation to the exchange team and my professors at the FAU in Germany, whose efforts paved the way for this exchange, culminating in the research presented in this thesis.

A special note of gratitude is reserved for Professor Jefferson, whose commitment, extensive guidance, and counsel were pivotal in shaping the trajectory of my project. Since I started studying at UFSC, Professor Jefferson has always been available to support every step of this research and my studies.

Furthermore, I wish to extend my appreciation to the examination board members for the opportunity to present my thesis. Their critical evaluation and feedback have been instrumental in validating this work.

Moreover, I am very grateful to my mother, brother, and father, whose support made my stay in Brazil possible. Their encouragement and belief in me have been a constant source of motivation. Furthermore, I would like to outline the help of my father in correcting this work.

I would also like to thank my extended family, colleagues, and friends, whose encouragement and support made it a pleasant time of my life.

To my beloved Ana Luíza, I deeply appreciate your emotional support, providing me with plenty of motivation throughout this journey. Your love and support have been critical in my determination to succeed.

RESUMO

Os problemas cardíacos como arritmias cardíacas representam uma das causas mais comuns de morte no mundo, arritmias cardíacas são originados em diferentes regiões do coração, apresentando, portanto, padrões muito específicos. O diagnóstico de doenças cardíacas é normalmente realizado por meio de eletrocardiogramas contando diferentes derivações. Esses exames são analisados por profissionais, entretanto, com o surgimento de melhores métodos de inteligência artificial, podemos observar um aumento nas tentativas de apoiar a análise com redes neurais. Existem várias abordagens utilizando tanto dados brutos quanto características desenvolvidas manualmente, como espectrogramas. As redes neurais mais populares são as redes neurais convolucionais que são frequentemente utilizadas como base, bem como as redes neurais recorrentes que também se mostram uma abordagem popular. A construção de um banco de dados para classificação é um desafio, por isso, utilizamos os dados do desafio Physionet 2021, composto por mais de 88.000 eletrocardiogramas disponíveis publicamente, provenientes de diversas fontes. O objetivo desta dissertação é apresentar uma visão geral e a fundamentação teórica dos conceitos mais importantes utilizados e necessários para a compreensão deste trabalho, bem como detalhar as configurações dos modelos, incluindo as estruturas ResNet e LSTMs, os diferentes hiperparâmetros e especificidades dos dados utilizados. Os resultados de diferentes modelos usando dados brutos ou espectrogramas e/ou LSTMs são descritos. O modelo com melhor desempenho é um ResNet34 baseado no uso de espectrogramas e 3 derivacoes, com uma exatidão total no teste de 78,48%, seguido por um ResNet34 que usa dados brutos e atinge 78,25%. Os modelos menores e mais simples baseados no ResNet18 também alcançaram acuracías muito altas, próximas aos modelos maiores. O melhor LSTM usando dados brutos alcançou apenas 56,91% no teste, com desempenho significativamente pior. Com base nas observações feitas, concluímos que um número menor de derivações já era suficiente e que os LSTMs não tiveram um bom desempenho, devido ao excesso de ajuste, mostrando as enormes vantagens dos modelos mais simples neste trabalho.

Palavras-chave: ECG, Electrocardiograma, Cardiologia, Arritmia Atrial, Classificação, Deep Learning, Redes Neurais

ABSTRACT

Cardiac arrhythmias and heart problems are among the most common causes of death worldwide, caused by various sicknesses in different heart regions and, therefore, having a pattern. The diagnosis of these diseases is commonly made with the help of electrocardiograms, which can include a different number of leads. Professionals usually analyze them; however, with the emergence of better artificial intelligence methods, there are more and more attempts to support the analysis with the help of deep neural networks. Various ideas exist for achieving that using raw data or handcrafted features such as spectrograms. For deep neural networks, convolutional neural networks are often used as a basis, and recurrent neural networks are a popular approach. Building an own database for classification is challenging, so for this work, the data of the Physionet 2021 challenge is used, consisting of over 88000 publicly available electrocardiograms with 12 leads from various sources. The overall goal of this dissertation is to create a well-performing, easy-to-use classifier for atrial arrhythmias. Therefore, an overview and theoretical background of the most important concepts being utilized is given needed to understand this work. Then, the model setups and configurations, including the ResNet Backbones and LSTM structures, are described, and the different hyperparameters and the specifics of the data are outlined. The results of different models using raw data or spectrograms and or LSTMs are given. The best-performing model is ResNet34 based, using Spectrograms and three leads having a full accuracy in the test of 78.48%, followed by a ResNet34 using raw data achieving 78.25 %. The smaller, simpler models based on ResNet18 also achieved very high accuracies close to the bigger models. The best LSTM using raw data only achieved 56.91 % in the test performed significantly worse. From the observations, we conclude that a smaller number of leads was already sufficient and that the LSTMs did not perform well, most likely due to overfitting showcasing the huge advantages of the simpler models in this work.

Keywords: ECG, Electrocardiogram, Cardiology, Atrial Arrhythmia, Classification, Deep Learning, Neural Networks

ZUSAMMENFASSUNG

Herzrhythmusstörungen und Herzprobleme gehören weltweit zu den häufigsten Todesursachen, verursacht in unterschiedlichen Herzregionen mit spezifischen Mustern. Die Diagnose dieser Krankheiten wird in der Regel mit Hilfe von Elektrokardiogrammen gestellt, die mit unterschiedlichen Ableitungen erzeugt werden können. Mit dem Aufkommen besserer Methoden der künstlichen Intelligenz gibt es jedoch immer mehr Versuche, die Analyse mit Hilfe von neuronalen Netzen zu unterstützen. Es gibt verschiedene Ansätze, um dies mit Hilfe von Rohdaten oder handgefertigten Merkmalen wie Spektrogrammen zu erreichen. Für neuronale Netze werden oft Faltungsnetze als Grundlage verwendet wie auch rekurrente neuronale Netze. Der Aufbau einer eigenen Datenbank für die Klassifizierung ist schwierig, daher werden für diese Arbeit die Daten der Physionet 2021 Challenge verwendet, die aus über 88000 öffentlich verfügbaren Elektrokardiogrammen mit 12 Ableitungen aus verschiedenen Quellen bestehen. Das übergeordnete Ziel dieser Arbeit ist es, einen gut funktionierenden und einfach zu verwendenden Klassifikator für atriale Arrhythmien zu entwickeln. Um diese Arbeit zu verstehen, werden zunächst ein Überblick und ein theoretischer Hintergrund zu den verwendeten Konzepten gegeben. Anschließend werden die Modelle, einschließlich der ResNet-Backbones und LSTM-Strukturen, beschrieben und die verschiedenen Hyperparameter sowie die verwendeten Daten skizziert. Die Ergebnisse der verschiedenen Modelle, die Rohdaten oder Spektrogramme und/oder LSTMs verwenden, werden angegeben. Das leistungsfähigste Modell ist ein ResNet34 auf der Grundlage von Spektrogrammen und drei Ableitungen mit einer Gesamtgenauigkeit von 78,48 % im Test, gefolgt von einem ResNet34, das mit Rohdaten 78,25 % erreicht. Die kleineren, einfacheren Modelle auf der Grundlage von ResNet18 erreichten ebenfalls sehr hohe Genauigkeiten. Das beste LSTM, das Rohdaten verwendet, erreichte im Test nur 56,91 % und schnitt damit schlechter ab. Aus den gemachten Beobachtungen können wir schließen, dass eine geringere Anzahl von Ableitungen bereits ausreichend war und dass die LSTMs für diesen Fall nicht gut abschnitten, was die enormen Vorteile der einfacheren Modelle in dieser Arbeit verdeutlicht.

Schlüsselwörter: EKG, Elektrokardiogram, Kardiologie, Atriale Arrhythmie, Klassifikation, Deep Learning, Neuronale Netze

RESUMO EXPANDIDO

Introdução

Problemas cardíacos, como arritmias cardíacas, representam uma das causas mais comuns de morte no mundo. As arritmias cardíacas se originam em diferentes regiões do coração e, portanto, apresentam padrões muito específicos. O diagnóstico de doenças cardíacas geralmente é feito com eletrocardiogramas com diferentes derivações. Esses eletrocardiogramas são analisados por profissionais; no entanto, com o surgimento de melhores métodos e desempenho de inteligência artificial, podemos observar um aumento nas tentativas de apoiar a análise com redes neurais artificiais. Há várias abordagens que usam dados brutos ou carateristicas extraídas dos sinais de eletrocardiograma, como espectrogramas. Uma das redes neurais mais populares são as redes neurais convolucionais em suas diferentes formas, que são frequentemente usadas como base, bem como as redes neurais recorrentes, que também estão se mostrando uma abordagem popular. A criação de um banco de dados para classificação é um desafio, por isso usamos os dados disponíveis gratuitamente do desafio Physionet 2021, que consiste em mais de 88.000 eletrocardiogramas disponíveis publicamente de várias fontes.

Objetivos

O objetivo desta dissertação é criar e testar pipelines de classificação capazes de processar eletrocardiogramas com números variados de derivações e diferentes durações de forma significativa para classificar sinais de acordo com suas respectivas arritmias, incluindo sinais sem arritmias. Isso inclui testar as diferentes características de dados brutos (o eletrocardiograma original) e espectrogramas (o eletrocardiograma processado).

Metodologia

Depois de explicar os conceitos mais relevantes necessários para a compreensão da dissertação, como as funções cardíacas básicas, o processamentos de sinal e os conceitos de aprendizagem de máquina profundo usados, os experimentos realizados são descritos em detalhes.

Primeiro, os conjuntos de dados são apresentados, incluindo uma descrição de seus parâmetros e o número de exemplos das diferentes arritmias atriais. Além disso, é

realizada a divisão entre 60% de dados para treinamento e 20% de cada conjunto para validação e teste.

Também, as configurações dos modelos são explicadas e delineadas. Os modelos têm um backbone baseado na ResNet 18 e na ResNet 34, que, em sua configuração, foram preparadas para processar eletrocardiogramas com uma janela de 5 segundos. Isso é implementado para usar apenas os primeiros cinco segundos para treinar apenas com o backbone, ou um LSTM de camada única de última geração é anexado para somar as janelas segmentadas do sinal.

Por fim, todos os hiperparâmetros são descritos. Isso inclui o pré-processamento da amostra, com uma frequencia de amostragem de 250 Hz e a filtragem digital passabanda entre 0,5 e 50 Hz, bem como a forma como os espectrogramas foram calculados. Também são descritos os parâmetros de treinamento, como os otimizadores Adam usados e os índices de aprendizado.

Resultados e Discussão

O modelo com melhor desempenho é baseado num ResNet34 no uso de espectrogramas e três derivações, com uma exatidão total no teste de 78,48%, seguido por um ResNet34 que usa dados brutos e atinge 78,25%. Os modelos menores e mais simples baseados no ResNet18 também alcançaram acuracías muito altas, próximas aos modelos maiores. O melhor LSTM usando dados brutos alcançou apenas 56,91% no teste, com desempenho significativamente pior.

Com base nos resultados, pode-se discutir que um número menor de derivações de eletrocardiograma já alcançará os melhores resultados possíveis. Além disso, observou-se que o aumento do tamanho da rede neural só proporcionou pequenas melhorias no desempenho, que mostra que os modelos menores desempenham satisfatoriamente. Foi possível observar na avaliação comparativa das janelas de toda a amostra, com o uso de apenas os primeiros cinco segundos para treinamento funcionou adequadamente, pois a exatidão foi similar em ambos os casos.

Analisando os resultados obtidos usando espectrogramas com diferentes tamanhos de janela, pudemos ver que 300 ms teve o melhor desempenho. Os diferentes tamanhos de janela apresentaram desempenho variável, de modo que uma otimização adicional pode ser vantajosa.

Além disso, poderíamos supor que os LSTMs cairam num overfit, pois a avaliação dos LSTMs com os backbones menores obteve melhores resultados do que os backbones maiores, embora a precisão do treinamento não tenha mostrado isso.

Considerações Finais

Com base nas observações feitas, concluímos que um número menor de derivações com três derivações já é suficiente quando usado só com os ResNets para crair um sistema classificador de arritmias a partri do eletrocardiograma. Como os LSTMs não tiveram um bom desempenho, concluimos que eles não foram adequados nesses experimentos mostrando, neste trabalho, as vantagens dos modelos mais simples de classificadores neurais.

Para trabalhos futuros, pode-se sugerir uma busca mais completa e refinamentos de hiperparâmetros para a criação de espectrogramas, a criação de um classificador que integre espectrogramas e dados brutos para usar ambas as caracteristicas, o teste de uma abordagem mais eficiente para otimizar uma janela móvel adequada para a classificação com ResNets, bem como a busca de melhores hiperparâmetros para treinar as LSTMs. Mais classificadores neurais para identificar outras patologias poderiam ser treinados com o mesmo pipeline. Para implementar os resultados desta pesquisa na vida real, é necessário definir uma interface comum para transmitir os dados para o classificador, bem como fazer testes reais para demonstrar a usabilidade e a aplicabilidade desta abordagem.

Palavras-chave: ECG, Electrocardiograma, Cardiologia, Arritmia Atrial, Classificação, Deep Learning, Redes Neurais, LSTM, ResNet

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ADC	Analog to Digital Converter
CinC	Computing in Cardiology
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
ECG	Electrocardiogram
FFT	Fast Fourier Transform
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
LSTM	Long-Short-Term Memory
RAM	Random-Access Memory
ResNet	Residual Network
RNN	Recurrent Neural Network
STFT	Short Time Fourier Transform

CONTENT

1 INTRODUCTION

1.1 OVERVIEW OF THE SUBJECT

It is well-known that arrhythmias and heart problems are among the most common causes of death worldwide (Virani et al., 2021). They can be caused by various sicknesses and abnormalities and have a very specific pattern (Sattar et al., 2023). Diagnosing these diseases is commonly done with the help of electrocardiograms (ECG), as they make various arrhythmias visible, showing very specific patterns in the electric signal emitted by the heart.

The heart's electrical activity (Electrocardiography) can be measured with different numbers of leads; however, using full 12-lead electrocardiograms is a widely used standard (Kligfield 2002). Nevertheless, signal measurement with fewer leads is also performed due to simplicity and lower costs. These subsets are also valuable but do not always capture the same amount of information as 12 leads (Aldrich et al. 1987, Drew et al. 2002, Green et al. 2007).

Electrocardiograms are usually analyzed by professional medical personnel. However, with the emergence of better artificial intelligence methods, there are more and more attempts to support data analysis with deep neural networks to achieve automated machine-based classification, as will be described in this thesis.

As input to the classifier, raw data is a popular option (Zhu et al. 2021; Singstad et al. 2020; Mostayed et al. 2018; Wong et al. 2021; Xu et al. 2020; Chen et al. 2019, Natarajan 2020) however we also find a lot of handcrafted features many times based on frequency components such as Mel-Frequency-Cepstral-Coefficients (Arpitha et al. 2021) or spectrograms (Huang et al. 2019, Gupta et al. 2021) are chosen. For data classification, methods such as support vector machines (Arpitha et al. 2021), k-nearest-neighbor (Arpitha et al. 2021, Ignacio et al. 2020), or Random Forest classifiers (Nonaka et al. 2020) are still popular options giving very good results depending on the type of data and features, however, with huge amounts of data, they are often outperformed by deep neural networks (DNN) as well as DNNs not depending on creating specialized features (Mahapatra 2018). For the latter, convolutional neural networks (CNN) are frequently used as a basis (Zhu et al. 2021; Singstad et al. 2020; Wong et al. 2021; Xu et al. 2020; Chen et al. 2019; Huang et al. 2019). In addition, the use of recurrent neural networks (RNNs) like different versions of Long-Short-Term

Memory units (LSTM) (Mostayed et al. 2018; Wong et al. 2021; Xu et al. 2020; Chen et al. 2019) is a well-liked approach to deal with the time dependence of electrocardiograms.

1.2 DATABASE: THE PHYSIONET/COMPUTING IN CARDIOLOGY CHALLENGE 2021

Since building an own database for classification is difficult, with a lot of legal requirements and being time-consuming, this work will be based on the data and builds on the idea of the Physionet 2021 challenge (Reyna et al. 2020, PhysioNet/Computing in Cardiology Challenge 2021) "Will Two Do? Varying Dimensions in Electrocardiography", since it has a large publicly available database.

The challenge is the follow-up to the ones in 2017 classifying single-lead electrocardiograms and 2020 (Perez et al. 2020) classifying 12-lead electrocardiograms. The emphasis is identical to the one of 2020, but as the title "Will Two Do? Varying Dimensions in Electrocardiography" already displays, it differs in the number of leads considered. Therefore, not only 12 leads but also a more limited number of leads was to be considered (Reyna et al. 2020, PhysioNet/Computing in Cardiology Challenge 2021) to build a fully automated classifier that could be trained and evaluated while still achieving good performances (Reyna et al. 2020, PhysioNet/Computing in Cardiology Challenge 2021).

The PhysioNet/Computing in Cardiology Challenge 2021 data corpus consists of over 88000 publicly available electrocardiograms with 12 leads from various sources displaying various arrhythmias (Reyna et al. 2020, PhysioNet/Computing in Cardiology Challenge 2021). Thirty-nine teams participated officially, and the winners were announced on 15 September 2021 at the Computing in Cardiology Conference in Brno, Czech Republic. The winning team was called ISIBrno-AIMT with an attention-based approach.

1.3 STATE OF RESEARCH

If we make a deep structured analysis of the literature, we find a lot of different approaches used to classify electrocardiograms, as will be shown here.

First, 53 articles were published on the program's website (PhysioNet/Computing in Cardiology Challenge 2021: Program) from different teams participating in the challenge, of which not all were published in a scientific journal.

Therefore, a complete online literature search was conducted. The search key included the three years with Physionet challenges with electrocardiograms and trying to catch all other approaches with more general keywords. The key "electrocardiogram" OR (("CinC" OR "Computing in Cardiology" OR "Physionet") AND ("2017" OR "2020" OR "2021") AND "challenge") AND ("automated" OR "neural network") AND ("classification" OR "detection") was chosen and applied to the online search site PubMed. A total of 90 results were found on the 7th of December 2022, with a time frame of five years. Of these 90 results, 48 remained after reviewing the titles and abstracts.

Summarizing the results, we can group them into different approaches. One of the very popular approaches is using different convolutional neural networks (Romdhane et al. 2020, Hsieh et al. 2020, Tutuko et al. 2021, Wu et al. 2020, Antoni et al. 2022, Krasteva et al. 2021, Zhang et al. 2021, Alsaleem et al. 2022, He et al. 2021, Lee et al. 2021, Nasim et al. 2022, Sbrollini et al. 2022, Jekova et al. 2022, Jiang et al. 2020, Prabhakararao et al. 2022, Aublin et al. 2022, Srivastava et al. 2022, Rubin et al. 2018) coming in different variations based on state-of-the-art architectures such as ResNet, DenseNet or Inception-based networks. However, we can sometimes also find simple convolutional layers. Another (convolutional) approach is squeeze-and-excitation networks (Ge et al. 2021, Yang et al. 2021, Zhu et al. 2021, Xu et al. 2022, Zhao et al. 2022). Recurrent neural network structures such as LSTMs or GRUs can also be found (Sawant et al. 2022, Cheng et al. 2021, Wang 2021, Xiong et al. 2018, Ping et al. 2020, Liang et al. 2020, Yildrim et al. 2020, Kang et al. 2022) as they provide simple solutions for inconsistent signal lengths and time dependencies. Recurrent models were often combined with an overlay convolutional neural network to build features, which were then processed by the recurrent units.

Another interesting approach is using attention-based models (Jiang et al. 2021, Gao et al. 2020, Xu et al. 2022), which seem to achieve high accuracy. Model ensembles are also a highly effective approach, as shown by the examples found (Khamis et al. 2018; Warrick et al. 2018; Rizwan et al. 2018). Some articles could not be sorted into any of the above categories as they, for example, used decision trees

(Shao et al. 2018), a sparse decomposition over composite dictionary (Raj et al. 2018), a canonical discriminant correlation analysis (Shi et al. 2021), a multilevel binary classifier (Mukherjee et al. 2019) or an adversarial neural network (Zhang et al. 2021).

For retrieving features, many researchers use raw signals as input data; however, handcrafting different features is also applied. Some specialized ways to extract features were found; however, the most commonly used types are wavelet-based features (Sharma et al. 2019, Toma et al. 2022, Bortolan et al. 2021, Kalidas et al. 2019, Hernandez et al. 2018) or frequency analysis-based-spectrograms (Jeong et al. 2021, Wickramasinghe et al. 2022).

1.4 RESEARCH QUESTION

This thesis aims to create a classification pipeline capable of handling electrocardiograms in a meaningful way to distinguish signals according to their respective arrhythmias, including signals of a healthy.

More specifically, after giving an overview of the most important concepts needed for understanding the approach, this work will explore options for classifying electrocardiograms with one or multiple arrhythmias, as there may be more than one label found for each electrocardiogram. Therefore, a classification system is created to analyze only a few seconds of a signal as a basis for a strong backbone on which to build.

After that, a network capable of handling time sequences of different lengths has to be raised, as sequence lengths, arrhythmia occurrences, and the number of specific events in electrocardiograms can vary. This creates the need to be able to analyze the entire signal to get stable and valid results.

Different data inputs are identified using raw signals or a frequency spectrum, giving different options with different characteristics. Frequency analysis has more information but is also more complex because the computation of the spectrogram is necessary, and the dimension is normally higher. Different model sizes need to be explored to show the effect of size, considering complexity, runtime, and accuracy. A special focus will be laid on making the model as efficient as possible while trying to stay as effective as possible to give a good guideline to show which size is needed.

The ECG data also provides the unique opportunity to explore different lead options using varying leads, as each lead carries different information, which may be

redundant. Therefore, knowing how many leads are needed to achieve good results is important.

Trying to summarize the subject within a single question, we state:

How well is a state-of-the-art, as easy as possible-to-use classification network with neural networks as defined in this thesis, capable of analyzing electrocardiograms for a given number of arrhythmias and leads using a raw signal or frequency analysis?

2 THEORETICAL BASIS

The purpose of this part is to provide an overview of the most important concepts that are needed to understand this thesis.

2.1 ELECTROCARDIOGRAMS

2.1.1 History

The history of ECGs starts with the first recordings of the electric impulses of a human heart with a mercury capillary electrometer by Augustus Waller in 1887. However, the tracings were of low quality. Willem Einthoven began studying ECGs with the mercury capillary electrometer and was already able to improve the recording quality before the turn of the century. Later, he used a string galvanometer as a design element, enhancing the quality such that in 1902, he recorded the first clinically applicable ECG. However, Einthoven is not only known for pioneering the development of the technology, but he also developed a system of electrocardiographic standardization that still is in use as a standard all over the world, introducing the triaxial bipolar system with three limb leads (Barold 2003).

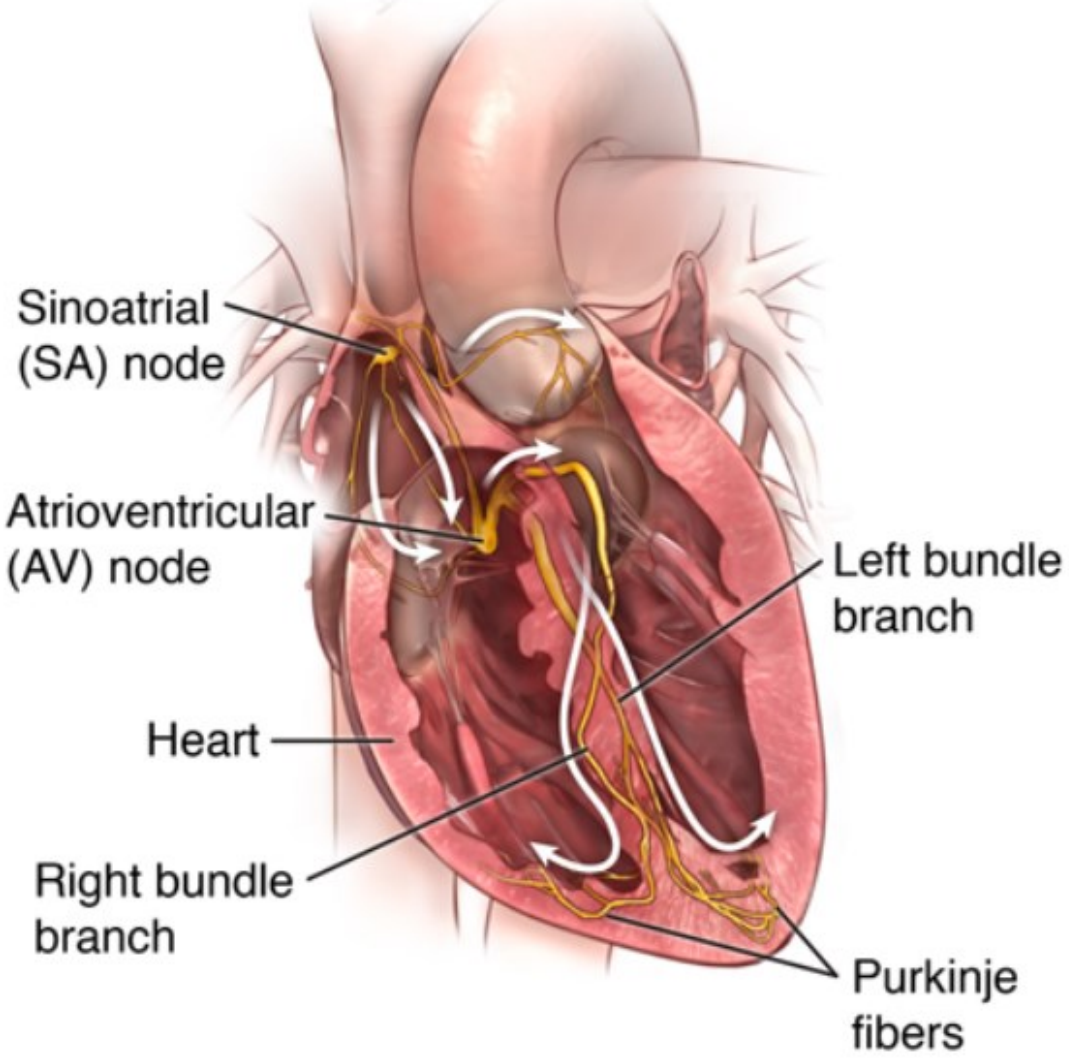
2.1.2 General Description

ECGs represent an electrical tracing of the heart and are recorded non-invasively from the body's surface. They are a very important diagnostic tool needed to investigate cardiovascular diseases. They are also used for monitoring patients treated with antiarrhythmics and other medicines, preoperative assessment of patients undergoing non-cardiac surgery, and screening individuals in high-risk occupations and sports (Sattar et al., 2023).

The heart is a muscular organ containing four chambers with two atria opening into the two ventricles. The wall consisting of muscle separating the four chambers is called the septum. The coronary arteries are on the surface of the heart, giving vascular supply to different heart regions. The heart also emits electrical signals. Their distribution is significant, as a 12-lead ECG can assess those and, thus, helps to localize and diagnose ischemic or infarcted areas (Sattar et al., 2023).

An overview of the structure of the electric systems of the heart is given in Figure 1.

Figure 1 – Electrical Systems of the Heart



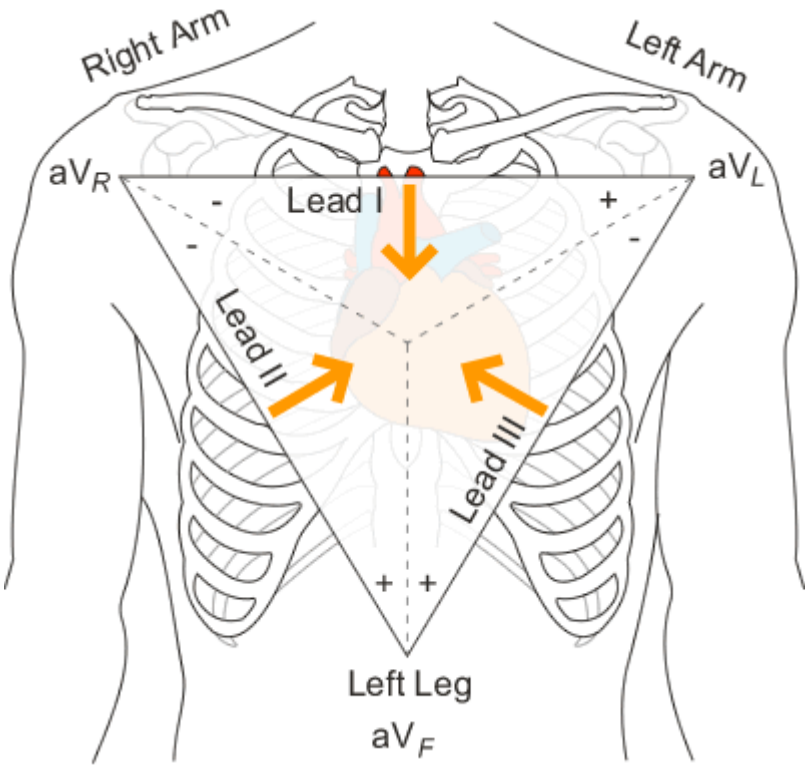
Source: https://www.hopkinsmedicine.org/health/conditions-and-diseases/anatomy-and-function-of-the-hearts-electrical-system

2.1.3 Measurement and Signals

The standard ECG consists of 12 leads divided into two groups - the limb leads and precordial leads. Limb leads consist of standard bipolar limb leads (I, II, and III) and augmented unipolar leads (aVL, aVF, and aVR). The bipolar limb leads and augmented unipolar leads are according to the Eindhoven triangle shown in Figure 2. The precordial leads consist of leads V1 to V6. The correct placement of all the electrodes for the 12 leads can be seen in Figure 3. The pattern of the limb lead shows the heart activity in a vertical and the precordial leads in a horizontal plane representing

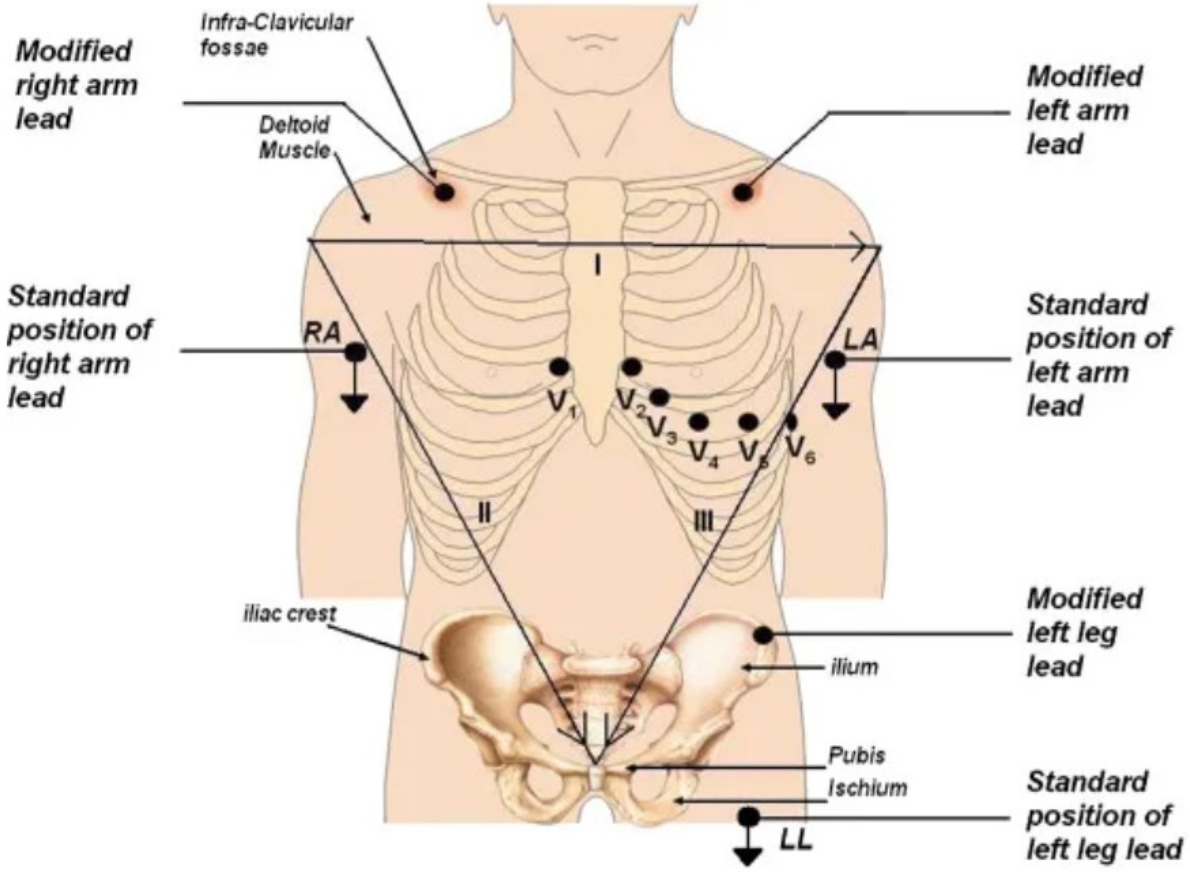
the electrical cardiac activity. The principle behind this is that depolarization moving to an electrode will get recorded as a positive deflection, while when it travels away from the electrode, it will show as a negative deflection (Sattar et al., 2023).

Figure 2 – The Eindhoven Triangle, Including Leads I, II, III, AVR, AVL, and AVF



Source:
https://www.nottingham.ac.uk/nursing/practice/resources/cardiology/function/bipolar_leads.php

Figure 3 – Placement of Electrodes on the Human Body (The Positions of the Arm and Leg Electrodes are at the Wrists and Ankles, Respectively)



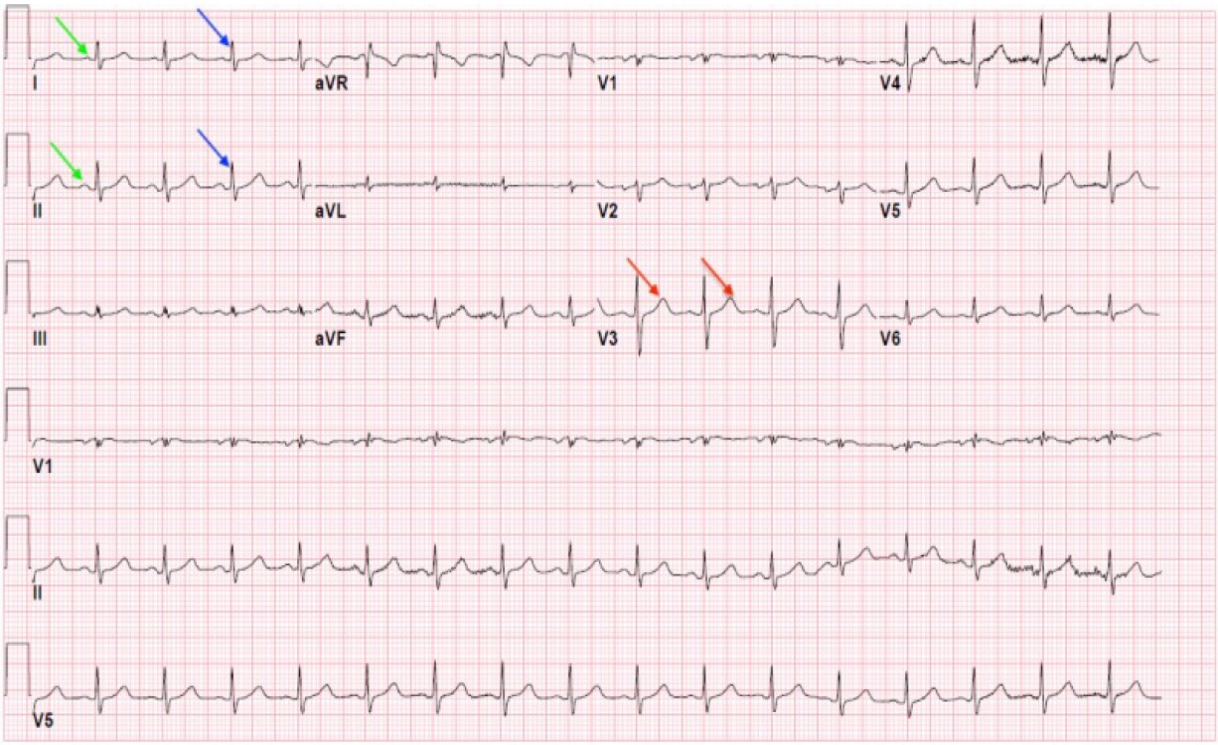
Source: Time Of Care: https://www.timeofcare.com/ecg-leads-placement-and-their-deflection-on-paper/

The first wave is the P-wave shown by the green arrow in Figure 4, which represents atrial depolarization initiated by the sinus node on the ECG. The PR interval is the interval from the beginning of atrial depolarization to the start of ventricular depolarization, which includes the delay at the atrioventricular node. The QRS complex (blue arrow in Figure 4) represents ventricular depolarization as electrical current passes down the atrioventricular node. The Q-Wave shows the depolarization of the interventricular septum, while the R-Wave, the tallest of the QRS complex, shows the stimulus passing through the ventricles during depolarization. The S wave is the last of the complex and visualizes the final depolarization of the Purkinje fibers. The last wave is the T-Wave (red arrow in Figure 4), depicting the ventricular repolarization. The ST segment represents the end of ventricular depolarization and the beginning of ventricular repolarization. The QT interval represents the start of depolarization to the

end of the repolarization of ventricles (Sattar et al., 2023). A fully traced 12-lead ECG is depicted in Figure 4.

All the waves and intervals have specific durations and heights, which are then interpreted by a medical professional, as any deviation from the norm can point out a cardiac abnormality (Sattar et al., 2023).

Figure 4 – Full 12-Lead Electrocardiogram (Green Arrow: points to P-Wave, Blue Arrow: points to QRS-Complex, Red Arrow: points to T-Wave)



Source: (Sattar et al. 2023)

2.2 CARDIAC ARRHYTHMIAS

According to the American Heart Association, "arrhythmia" refers to any problem in a person's heartbeat rate or rhythm. During an arrhythmia, the electrical impulses may be too fast, slow, or erratic, causing an irregular heartbeat (American Heart Association, 2023).

Examples of arrhythmia are Atrial Fibrillation, Atrial Flutter, Bradycardia, Conduction Disorders, Premature Contractions, and Tachycardia, among others (American Heart Association, 2023).

The normal sequence of a heart contraction starts when an electrical impulse is created in the sinus node and moves through the heart. After beginning in the right

atrium, the signal spreads through the atria to the atrioventricular node. From there, the impulse goes to the Bundle of His and through the Purkinje fibers, creating the ventricles' contraction. This heartbeat sequence is normally very regular with organized contraction (American Heart Association, 2023).

If any change from the normal sequence of electrical impulses occurs, it is referred to as arrhythmia. Some of them are so brief that the overall function is barely affected. If they, however, occur frequently or for a longer duration, they can have a negative impact on the function of the heart (American Heart Association, 2023).

Even though the sinus node normally triggers the heart's pace, almost all heart tissue can start an impulse under the right conditions, leading to a heartbeat. If these other cells in the heart fire unwantedly and start the electrical activity, this might disturb the normal sequence of the heart, causing arrhythmia. Other typical causes include delays or blockages of electrical signals managing the beat, stress situations of a person, excessive exertion or strain, an imbalance of hormones or electrolytes, changes of heart tissue caused by, e.g. changes in blood flow, stiffening or scarring of the tissue or damage to the electrical system of the heart as well as some heart medications (American Heart Association, 2023).

2.3 SIGNAL PROCESSING AND FILTERING

2.3.1 Short-Time-Fourier-Transform

A standard approach to analysing a one-dimensional signal is to divide it into its frequency components. The Discrete Fourier Transform (DFT) describes the discrete conversion of a discrete function into the Fourier space, displaying the frequencies of the signals. It is defined as

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-i \cdot k \cdot \frac{2\pi}{N} \cdot n} \quad k \in Z$$

The Short Time Fourier Transform (STFT) is an analysis with the DFT of short potentially overlapping windows with a given size N of a signal with an FFT-size L and hop-size H and is defined as

$$X[k] = \sum_{n=0}^{L-1} \mathrm{x[n + mH]} \cdot \omega[\mathrm{n}] \cdot e^{-i \cdot k \cdot \frac{2\pi}{N} \cdot n} \quad k \in Z \,; L < N$$

As the DFT can only be used to analyze periodic signals as it does not portray any occurring frequency changes within the signal, the STFT was created. As the STFT is a DFT of small windows in the signal, it can make premises on frequency changes over time with a frequency resolution of $f_{resolution} = \frac{f_{samplerate}}{L}$ with frequencies being sorted into L bins, creating a 2D spectrogram displaying the frequency components in relation to time.

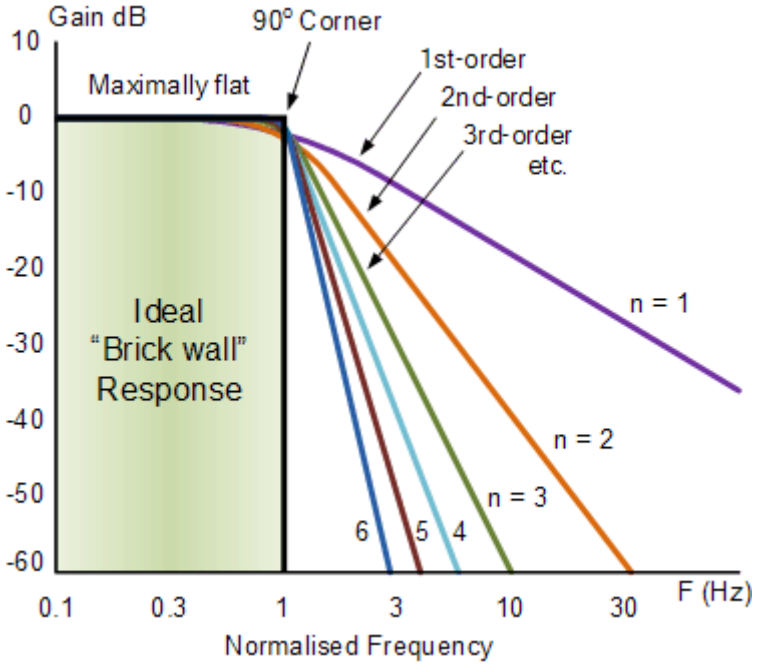
The important thing one has to consider is that the better the frequency resolution is, the lower the time resolution. The same applies to the other way around. This effect is caused when increasing window size, having more bins giving more frequency information and, therefore, a more accurate frequency resolution, while the frequencies can occur at any point in the larger window, decreasing time resolution. This is important when making a trade-off between high-frequency and high-time resolutions to serve the application's purpose best.

2.3.2 Frequency Filter for Signals

A frequency filter is a tool used to suppress a band of unwanted frequencies. The filters can be low-pass filters eliminating frequencies above a predetermined border, high-pass filters removing frequencies below a defined border, bandpass filters allowing only frequencies within a predefined range, and band-stop filters clearing frequencies in a predetermined range.

Those filters can be designed in several ways, like the Chebyshev or Bessel filter or the used Butterworth Filter. This filter, first described by Stephen Butterworth (Butterworth, 1930), has the advantage of a frequency response being as flat as possible in the passband and stopband (no ripples), giving very smooth and good results. For an ideal Butterworth Filter, the cutoff frequency would be at -3db, and depending on the order, the frequency will drop off 20db times the order after the cutoff every decade (Figure 5). It will have no ripples trying to approximate a 100 % cutoff for higher orders. In reality, this cannot be achieved, but the Butterworth filter is still a good approximation to filter frequencies very good.

Figure 5 – Filter Response of a Butterworth-Filter for Different Orders



Source: https://www.electronics-tutorials.ws/filter/filter_8.html

2.4 PERFORMACE MEASUREMENT

2.4.1 True Positives, False Positives, True Negatives, and False Negatives

The listed metrics in the headline are very important, describing other metrics and shall be defined here. A prediction being true positive (TP) defines that the prediction and label both are “true”, false negative (FN) that the prediction is “false” while the target is “true”, false positive (FP) that the prediction is “true” while the label is “false” and true negative (TN) both are “false”. This behavior is visualized in the confusion table (Table 1).

Table 1 – Confusion Table

		Prediction	
		1	0
Label	1	True Positive	False Negative
	0	False Positive	True Negative

Source: Created by the Author and based on Goutte and Gaussier 2005

2.4.2 Accuracy

For this thesis, two different accuracies are used. Assuming a target label and output vector of size m x n with m classes and n samples, the per-label accuracy is

calculated by counting all true positives and negatives and dividing them by all predictions (m x n) made.

The other accuracy we calculate is full-label accuracy, where we calculate not per class and per sample but only per sample. Therefore, we only consider a “true” if the whole label vector of a sample has been predicted correctly. To calculate this measurement, we divide all the correctly estimated label vectors by all the samples.

2.4.3 Recall, Precision, and F1-Score

Recall, Precision, and F-Score are other performance measurement methods frequently used to describe classifier performance. The Recall (R) is defined by $R = \frac{TPs}{TPs+FNs}$ and describes how many of the positive labels were predicted correctly. The Precision (P) can be calculated by the formula $P = \frac{TPs}{TPs+TNs}$ and describes how many of the positive predictions are, in fact, positive and correct. The F1-Score is a combination of both and is defined by $F1 = 2 \times \frac{P \times R}{R + 2 \times P} = \frac{2 \times TPs}{2 \times TPs + FNs + FPs}$ (Goutte and Gaussier 2005).

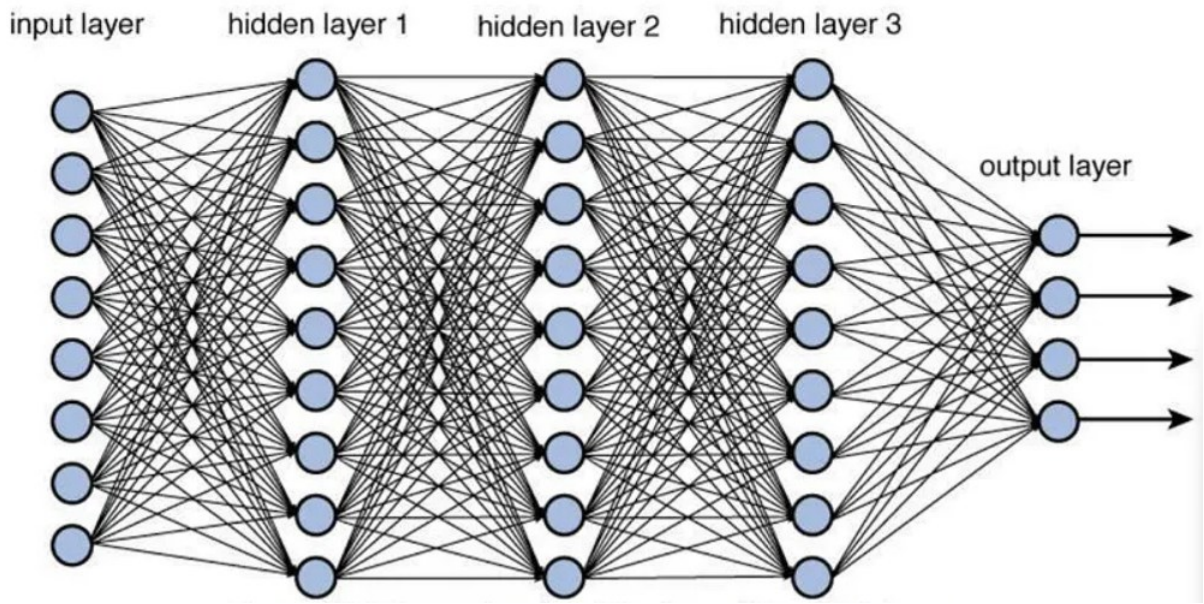
2.5 DEEP LEARNING ARCHITECTURES

Deep learning is a term used for the family of artificial neural networks and representation learning and its techniques. They have been used successfully in the last years for different classification and learning tasks in various fields, from computer vision to audio processing. The following will describe the relevant architectures and concepts for this thesis.

2.5.1 Deep Neural Networks

A deep neural network (DNN) has multiple layers, normally with two or more hidden layers. A hidden layer contains weighted nodes, which, multiplied with the layer's input, calculate a weighted representation (Goodfellow et al., 2016). An exemplary structure of a DNN, which in this case seems to be a fully connected neural network, can be seen in Figure 6. There are three main ways to train a network.

Figure 6 – Exemplary Structure of a DNN



Source: https://towardsdatascience.com/training-deep-neural-networks-9fdb1964b964

The two following passes do supervised training of the weights. The forward pass transmits the input through the layers, calculating an output. The output is then compared with the given labels, calculating a loss with a loss function like cross-entropy loss, showing how well the output represents the given labels (Goodfellow et al., 2016). After that, the backward pass calculates a gradient for each layer to optimize and minimize the loss. This starts at the last layer and is done layer by layer using the chain rule, whereas the adjusted error of the previous layer passing backwards can be utilized to calculate the error for the current layer, giving a gradient to improve (Goodfellow et al., 2016).

Unsupervised learning is an option when no labels are available. A very popular group of models to do unsupervised learning are autoencoders who create a hidden representation of the data and then try to reconstruct the original sample, calculating the loss between reconstruction and original. The learned representations can then be clustered into classes of representations of proximity (Goodfellow et al. 2016).

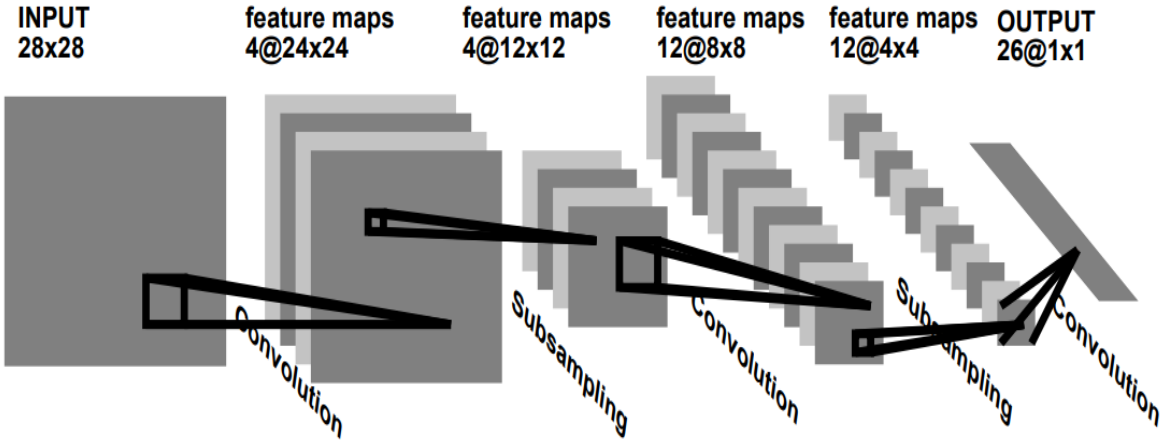
Semi-supervised training is a mixed form between the two other methods and can be very powerful against overfitting first initializing the weights with a large unlabeled dataset to find the best features representing the data to do then finetuning with a smaller labelled dataset (Goodfellow et al. 2016).

Optimization is then done with one of the gradient descent methods, which can be a simple stochastic gradient descent or more advanced methods building up on the same principle (Goodfellow et al., 2016).

2.5.2 Convolutional Neural Networks

A Convolutional neural network (CNN) is a deep learning concept created by LeCun et al. (1998) for handwritten letter recognition. It has proven to be a very efficient tool to be applied to shapes like images or spectrograms. The plan for the original network depicting its structure can be seen in Figure 7. Its design is based on traditional pattern recognition principles using convolutional layers with trainable convolutional kernels for feature learning and extraction and fully connected layers for classification. Within the convolutional layers, there are several important architectural approaches applied. They are first local receptive fields, second shared weights, and finally, spatial and temporal sub-sampling (pooling). This all increases the performance of architecture designed in such a manner.

Figure 7 – Network structure of the original CNN



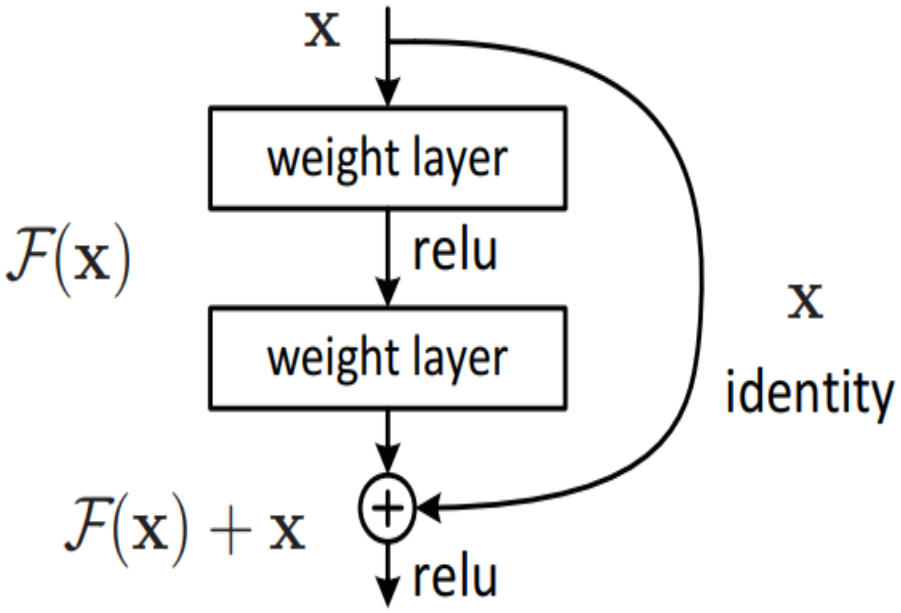
Source: Lecun et al. 1995

2.5.3 Residual Networks

Residual Networks were created to train deep neural networks to learn higher-level and more discriminate feature results in various optimization problems. For these, vanishing or exploding gradients or the degradation problem, which describes that deeper nets may not perform better despite their larger amount of trainable weights, were a limiting factor in network sizes being still efficient.

Therefore, He et al. (2016) evaluated a deep residual learning architecture named residual network (ResNet). It uses residual mappings to prevent learning and optimizing unreferenced underlying mappings H(x) to the input x, expressing it through a residual mapping $F(x) = H(x) - x$ to achieve a viable solution to counteract the degradation mentioned above problem successfully. A depiction of a residual block can be seen in Figure 8.

Figure 8 – Structure of a Basic Residual Block



Source: He et al. 2015

On top of that, He et al. evaluated several structure sizes and presented several different and typical ResNet architectures based on their size (number of concatenated layers), which proved their value in practical experiments.

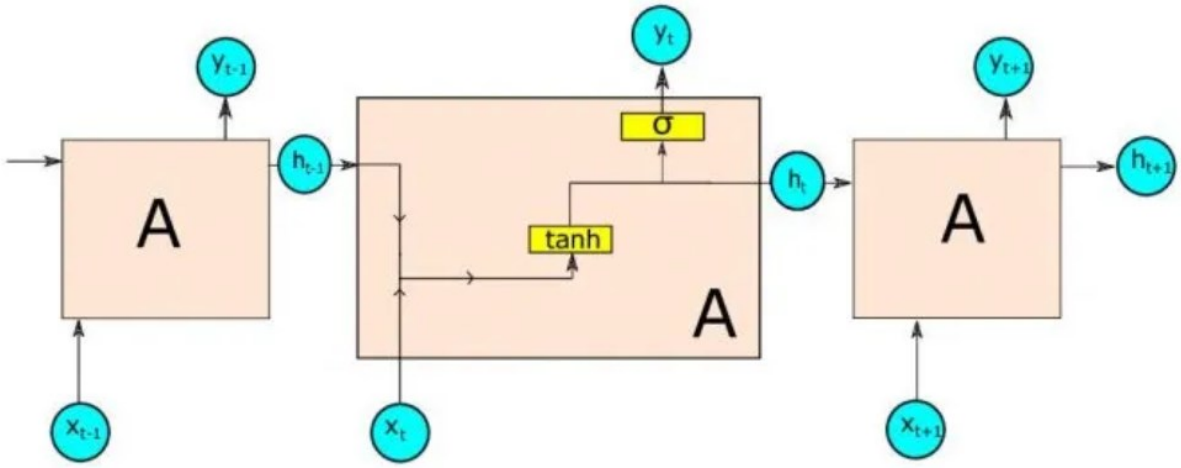
2.5.4 Recurrent Neural Networks

Recurrent neural networks (RNNs) were invented to process signals with a temporal context. Feeding a whole sequence into a big network is generally not a good idea due to inefficient memory usage, a difficult or even impossible training process, and no differences between spatial and temporal dimensions; it also does not work in real-time (Gehring et al., 2017).

One of the very first and simple RNNs was the Elman cell depicted in Figure 9 (Elman et al., 1990). There, we see that RNNs compute the output from the input and have an additional hidden state. Therefore, to compute the output, the input and the previous hidden state are calculated in an updated hidden state from which the output is determined and used as the initial hidden state for the next time step.

Training the network is done by a classical forward pass and the backpropagation through time, as each output also depends on the previous timesteps.

Figure 9 – Structure of the Elman Cell



Source: https://towardsdatascience.com/recurrent-neural-networks-part-1-498230290534

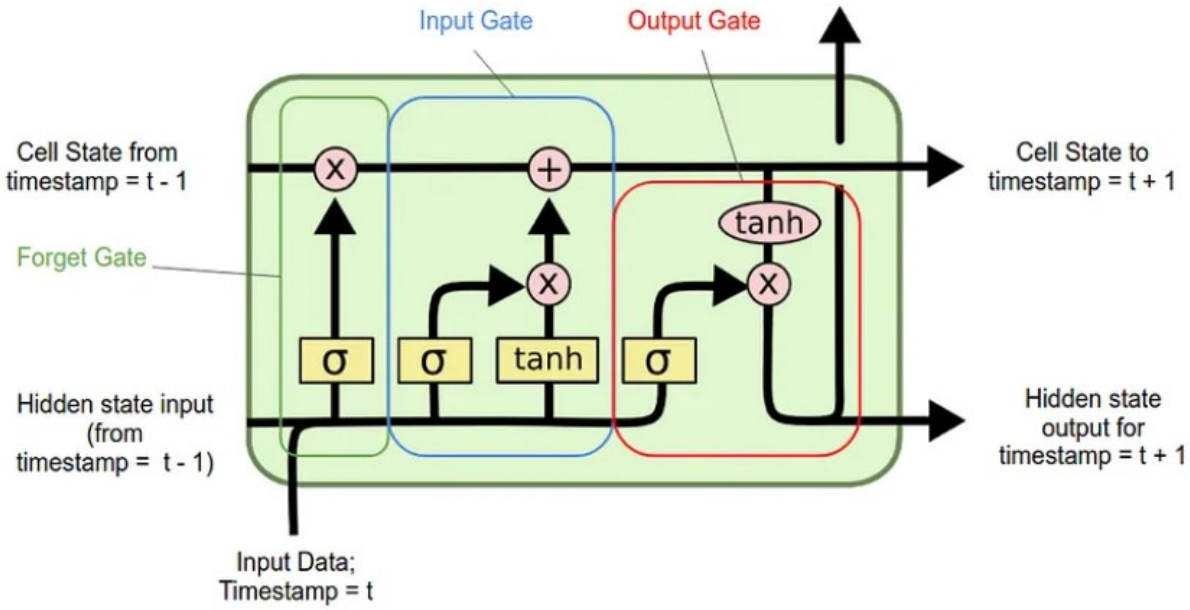
2.5.5 Long-Short-Term-Memory Units

A Long-Short-Term-Memory (LSTM) unit is a special RNN concept for better long-term dependencies and was created by Hochreiter and Schmidhuber (Hochreiter and Schmidhuber 1997).

Basic RNNs have the problem that they cannot effectively learn long-term dependencies. This is caused by the hidden state being overwritten each time, short-term dependencies hiding long-term dependencies due to their longer, exponentially smaller gradients, and exploding and vanishing gradients in training (Hochreiter and Schmidhuber 1997).

Therefore, the LSTM introduces an extra cell state. It is updated by the combined old hidden state and input through a forget gate deciding to delete long-term information and an input gate deciding on new input for the cell state. The new hidden state and output are then calculated with the new cell state, the old hidden state, and the input. An LSTM cell is visualized in Figure 10.

Figure 10 – Structure of an LSTM Cell



Source: https://medium.com/analytics-vidhya/lstms-explained-a-complete-technically-accurate-conceptual-guide-with-keras-2a650327e8f2

3 DATA SOURCES AND METHODS

This paragraph describes the used data and different experimental setups and parameters.

3.1 DATA

As mentioned in the introduction, the dataset used was compiled from the public data available in the PhysioNet/CinC 2021 challenge. Five of the seven sources had at least parts of their datasets publicly available (PhysioNet/CinC 2021).

The first source is data published during the China Physiological Signal Challenge 2018 at the 7th International Conference on Biomedical Engineering and Biotechnology in Nanjing, China. This source contains the officially used (CPSC in Table 3) and unused data (CPSC_Extra in Table 3) from this challenge. Together, they number 13,256 ECGs, of which 10,330 were shared and publicly available. The recordings were between 6 and 144 seconds with a sampling frequency of 500 Hz and were taken from hospitals.

The second dataset is the St Petersburg INCART 12-lead Arrhythmia Database (St Petersburg in Table 3). This set is comprised of 74 ECGs, which were all shared. The recordings are all 30 minutes long with a sampling frequency of 257 Hz.

Another source is from the Physikalisch-Technische Bundesanstalt, including their two public datasets (PTB in Table 3 and PTB_XL in Table 4). They, in total, number 22,353 ECGs. The clinical recordings are between 10 and 120 seconds long, with a sampling frequency between 500 and 1,000 Hz.

The fourth database is a Georgia database (Georgia in Table 4), mimicking the demographic reality of the Southeastern United States. This database contains 20,672 ECGs, of which 10,344 ECGs were shared publicly. The lengths of the recordings vary between five and 10 seconds with a sampling frequency of 500 Hz.

The last dataset coming from hospitals with publicly available data is taken from Chapman University, the Shaoxing People's Hospital (Chapman_Shaoxing in Table 4), and the Ningbo First Hospital (Ningbo in Table 4) databases. This dataset has a total of 45,152 ECGs, which are all publicly available. They have a length of 10 seconds each with a sampling frequency of 500 Hz.

Even though the dataset of St Petersburg was available, it is not used in this thesis as it only contains a few samples, which did not have a considerable overlap with the labels while having very different measurement parameters and the data being very long and difficult to handle.

The remaining datasets were annotated using SNOMED-CT codes. There are 133 codes, of which more than one could be true for a sample. The labels are very unevenly distributed, so for the challenge itself, only 26 labels with 30 different codes were used (four pairs were counted as the same classification). For this work, we focus on only eight labels of interest, all in the family of atrial arrhythmias. This reduction was made to facilitate training and the expressiveness of the results while only using labels with sufficient examples. Atrial arrhythmias were chosen over ventricular arrhythmias, as they are the most common (David Siu and Hanh Tse 2008).

Of the complete original database, we, therefore, include them in a total of 75412 samples showcasing our labels, with a total of 81764 observations being atrial fibrillation, atrial flutter, 1st-degree av block, premature atrial contraction, supraventricular premature beats being counted as the same label as premature atrial contraction, as even though they have different nomenclature codes they are the same, sinus arrhythmia, sinus bradycardia, and sinus tachycardia as well as the sinus rhythm. The number for each observation can be seen in Table 2, and the distribution of samples from the different datasets can be seen in Tables 3 and 4.

Table 2 – Evaluated Labels with Description (Blue Labels are counted as the same Label)

	SNOMED-CT Code	Abbreviation	Total
atrial fibrillation	164889003	AF	5255
atrial flutter	164890007	AFL	8374
1st degree av block	270492004	IAVB	3534
sinus rhythm	426783006	NSR	28971
premature atrial contraction	284470004	PAC	3041
sinus arrhythmia	427393009	SA	3790
sinus bradycardia	426177001	SB	18918
sinus tachycardia	427084000	STach	9657
supraventricular premature beats	63593006	SVPB	224
TOTAL			81764

Source: Based on https://github.com/physionetchallenges/evaluation-2021/blob/main/dx_mapping_scored.csv and modified by the author

Table 3 – Number of Observed Labels per Dataset (Blue Labels are counted as the same Label) (Part1)

	CPSC	CPSC_Extra	StPetersburg	PTB
atrial fibrillation	1221	153	2	15
atrial flutter	0	54	0	1
1st degree av block	722	106	0	0
sinus rhythm	918	4	0	80
premature atrial contraction	616	73	3	0
sinus arrhythmia	0	11	2	0
sinus bradycardia	0	45	0	0
sinus tachycardia	0	303	11	1
supraventricular premature beats	0	53	4	0
TOTAL	3477	802	22	97

Source: Based on https://github.com/physionetchallenges/evaluation-2021/blob/main/dx_mapping_scored.csv and modified by the author

Table 4 – Number of Observed Labels per Dataset (Blue Labels are counted as the same Label) (Part2)

	PTB_XL	Georgia	Chapman_Shaoxing	Ningbo
atrial fibrillation	1514	570	1780	0
atrial flutter	73	186	445	7615
1st degree av block	797	769	247	893
sinus rhythm	18092	1752	1826	6299
premature atrial contraction	398	639	258	1054
sinus arrhythmia	772	455	0	2550
sinus bradycardia	637	1677	3889	12670
sinus tachycardia	826	1261	1568	5687
supraventricular premature beats	157	1	0	9
TOTAL	23266	7310	10013	36777

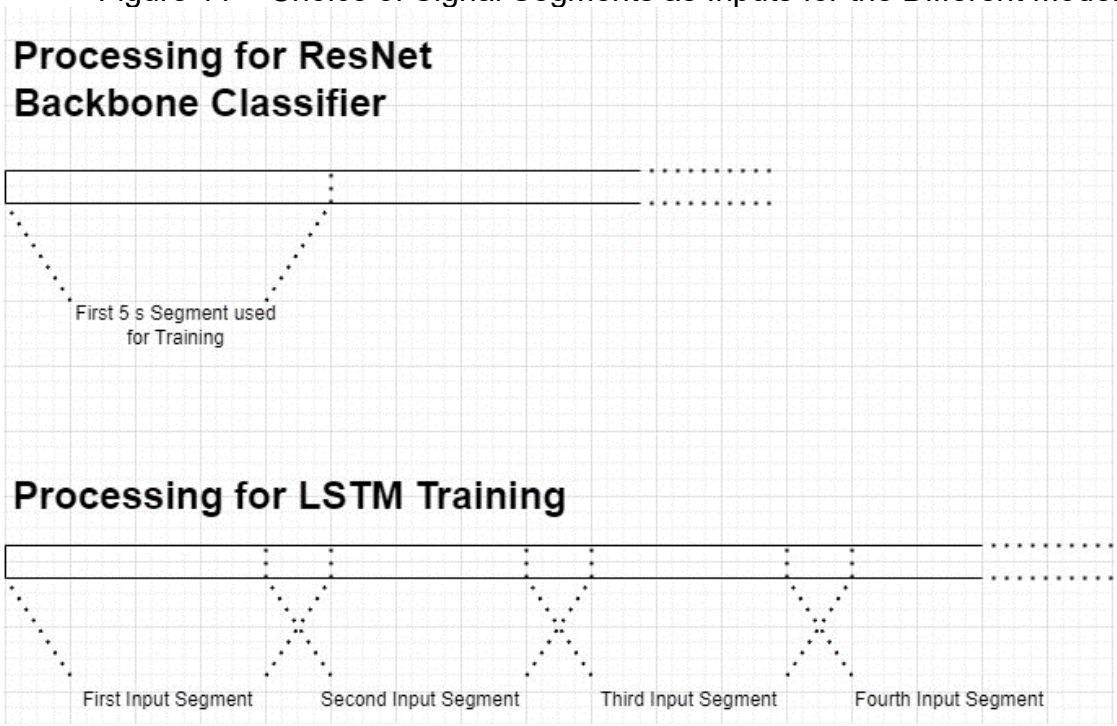
Source: Based on https://github.com/physionetchallenges/evaluation-2021/blob/main/dx_mapping_scored.csv and modified by the author

3.2 EXPERIMENTAL SETUPS

As input data, segments of five seconds duration are used. The length of 5 seconds was chosen as a tradeoff, as this length always includes several heartbeats with a great chance of fully including an occurrence of an arrhythmia while not being too difficult to handle. If the sample is longer, the whole sample can be processed with a moving window of the determined length. Furthermore, the length of the samples begins with this duration, which is also a practical aspect. If the training is done without LSTM, the first five seconds of the signal are used for training, validation, and testing,

as visualized in Figure 11, and an extra evaluation on the test set with a moving window with an overlap of 50 % was made to show performance on the whole sample. This evaluation was done in two ways. The first was to assume the sequence having the same label in all of the segments and evaluate each segment against the label vector, and the second was to sum all the classifications of all segments of a sample and evaluate it against its label vector. With LSTM training, segments overlapping one second are cut sequentially, as seen in Figure 11.

Figure 11 – Choice of Signal Segments as Inputs for the Different Models



Source: By the author

If raw data is used, the desired number of leads is selected, and the data is sampled to a common rate of 250 Hz. The rate of only 250 Hz was chosen to reduce the data without losing necessary information because all the important frequencies are within the range of 125 Hz, which would be the maximum included frequency according to the Nyquist theorem. Then, the Analog-to-Digital (ADC) gains and baselines are normed and filtered with a bandpass Butterworth filter of the 5^{th} order having a minimal frequency of 0.5 Hz and a maximum frequency of 50 Hz because frequencies of low interest in an ECG can be suppressed and higher frequency noise

and baseline drift filtered out. With that, the input signal parts have a length of 1250 bins.

If an additional frequency analysis is made, an STFT of different window sizes depending on the model is performed while implementing in the library suggested Hann window and using a hop size of 2 bins (8 milliseconds) is chosen to calculate the frequency spectrogram of the signal, which is then used as input. The frequency direction is then reduced, discarding the bin for the constant without frequency and the bins, which have a frequency above the filtered 50 Hz, reducing the number of bins in the frequency dimension to only include relevant data.

The numbers of leads chosen for the experiments are single lead (lead II), three leads (leads II, V1, V5), and twelve leads (leads I, II, III, aVR, aVL, aVF, V1, V2, V3, V4, V5, V6). The single lead II was chosen because it has shown great success in showing the majority of arrhythmias. For the three-lead choice, the leads, which represent atrial arrhythmias, are used. The idea of training on fewer leads is that making an ECG with fewer leads is simpler, and the data is faster and simpler to process due to its lower dimensionality. On top of that, a smaller number of leads already carries a significant part of the information of a full lead ECG, which can already be enough to classify arrhythmias perfectly.

Several neural networks are trained, all of them having a ResNet backbone. The two chosen sizes are ResNet18 and ResNet34, as they are still fairly small and fast to train with a relatively small risk of an overfit. Two adaptations were implemented as they were required to process two- and three-dimensional data to handle the different data types. The two-dimensional ResNets, as shown in Figure 12, have an initial convolutional layer with a kernel size of 13 in the time direction and one in the lead direction to handle the different leads separately. The ResNet blocks have kernel sizes of 9 in the time direction. The three-dimensional Nets, as shown in Figure 13, start with three-dimensional kernel sizes of 5x5 in the initial convolutional layer and 3x3 for the residual blocks chosen, while in the lead direction, the kernel size is 1 to keep the leads separated. All the data was padded circularly when necessary. From the input, 512 feature vectors were calculated for each ECG lead, which was then summed up by a convolutional layer to 512 feature vectors being subsequently transformed to 512 features with average pooling. Out of those 512 features, the labels

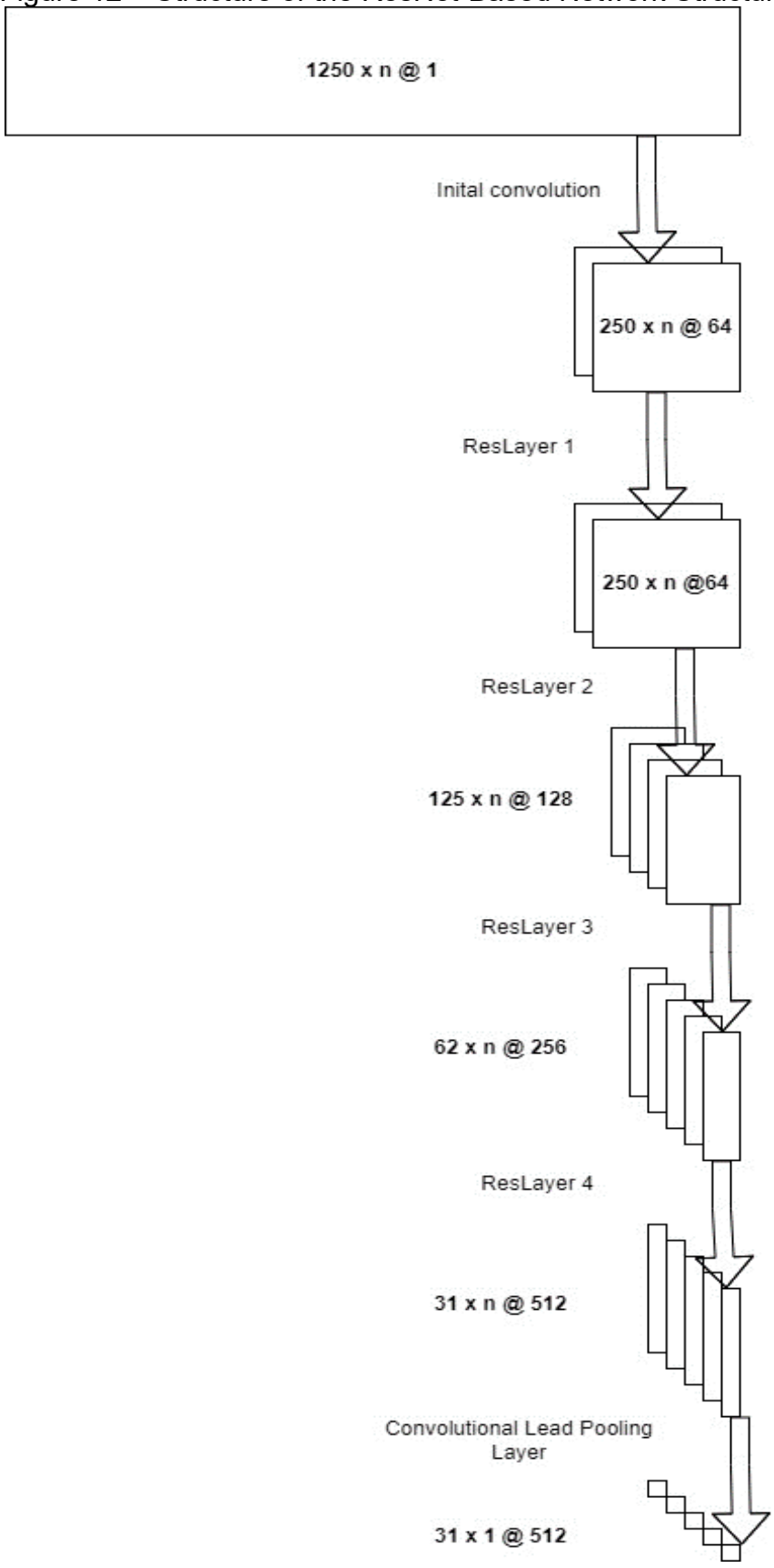
were calculated with fully connected and sigmoid layers to achieve an output between zero and one. We can see the visualization of the classifying structure in Figure 14.

Using an LSTM, the segments are first transformed to 512 features as described above and then put through the LSTM with cell state and hidden state also has the size of 512. The output was calculated just at the end of the sequence. The schematics of this net are visualized in Figure 15. The implementation of the LSTM is based on an exemplary implementation of an article explaining the basic LSTM and its implementation (Esposito 2020). Due to the samples having different lengths, the batch size must be one, or the data would have to be padded to (groups of) the same length, which only gave bad results in the first test, so this approach was discarded.

The exact parameters of all included models are highlighted with their respective results in the result section.

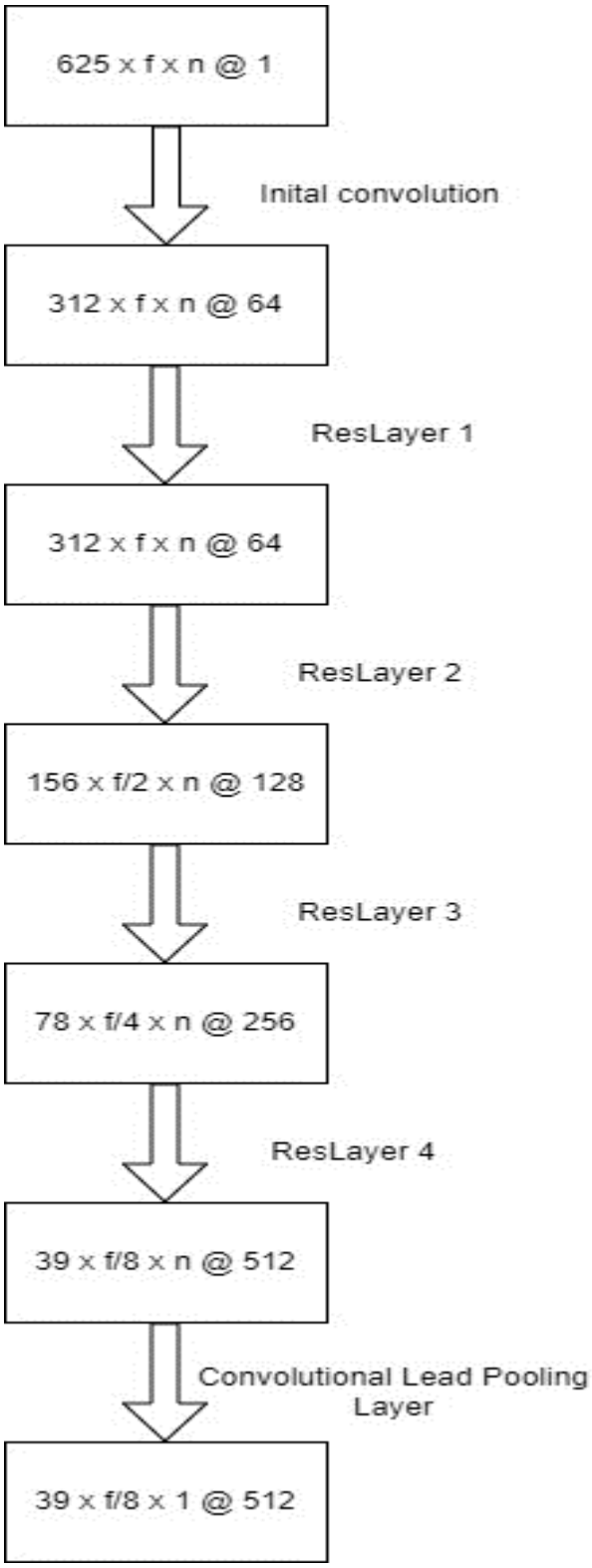
All implementations were written in Python 3, mainly using PyTorch's deep learning library and (standard) libraries like NumPy, math, pickle, tensorboardX, glob, scipy, and resampy. Some of the helping methods for loading data were used from implementing the Physionet challenge (Physionet/CinC 2021) from which the data originated.

Figure 12 – Structure of the ResNet-Based Network Structure



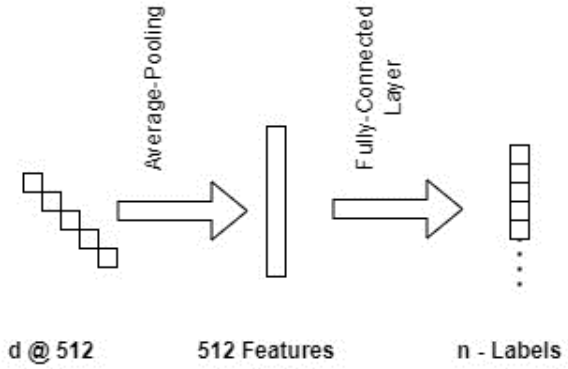
Source: Drawing of the Author

Figure 13 – Structure of the ResNet-Based Network Structure Using Spectrograms with a 200 ms window.



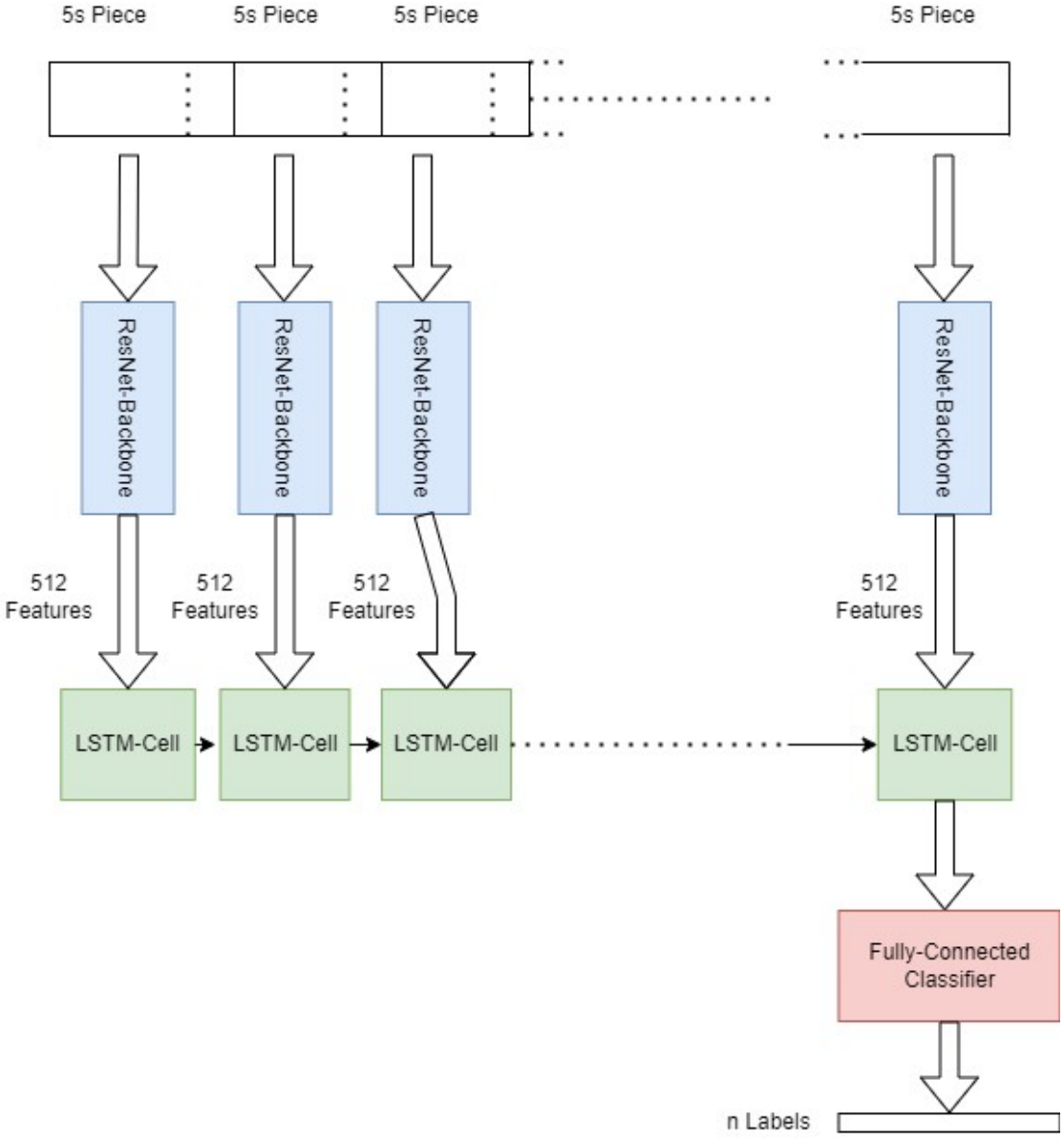
Source: Drawing of the Author

Figure 14 – Structure of the Used Basic Classifying Structure



Source: Drawing of the Author

Figure 15 – Structure of the Used LSTM-Based Network



Source: Drawing of the Author

3.3 EXPERIMENTAL PARAMETERS

Many models were trained to test parameters and debug the models and the classification pipeline. For this thesis, only the most recent and best working models shall be of interest, highlighting different parameters like batch sizes, learning rates, and model configuration and their effects.

A binary cross entropy loss is utilized to train the models due to being a multilabel problem. A commonly used Adam optimizer optimizes with betas of 0.9 and 0.999 and the AMS Grad extension switched on. The extension was used, as it was shown that Adam does not have a convergence guarantee, while the AMS Grad extension is supposed to fix this (Reddi et al. 2019). The learning rate is adapted to the model. The different models are implemented with an early stopping criterion, which stops the training after seven epochs without improvement and a learning rate decay of 50% after every three epochs without improvement in the loss.

The available data was split into 60 % training and each, 20 % test, and 20 % validation set. The splits are saved to guarantee the same split for all the training, making the models comparable. The exact numbers of samples from each dataset and label distribution for each split are given in Tables 5 and 6.

The training of models was and is mostly done on the free version of Google Colab because it features free GPU resources, accelerating the training considerably.

Table 5 – Absolut Number of Labels per Set

Absolut	Train	Validation	Test
atrial fibrillation	3208	996	1049
atrial flutter	5066	1639	1669
1st degree AV block	2123	759	652
sinus rhythm	17332	5858	5781
sinus arrhythmia	2291	760	737
sinus bradycardia	11266	3807	3845
sinus tachycardia	5794	1930	1922
premature atrial contraction and supraventricular premature beats	1949	642	665
TOTAL	49029	16391	16320

Source: Elaborated by the Author

Table 6 – Percentage of Labels per Set

In %	Train	Validation	Test
atrial fibrillation	61,07%	18,96%	19,97%
atrial flutter	60,50%	19,57%	19,93%
1st degree AV block	60,07%	21,48%	18,45%
sinus rhythm	59,83%	20,22%	19,95%
sinus arrhythmia	60,48%	20,06%	19,46%
sinus bradycardia	59,55%	20,12%	20,32%
sinus tachycardia	60,07%	20,01%	19,93%
premature atrial contraction and supraventricular premature beats	59,86%	19,72%	20,42%
TOTAL	59,98%	20,05%	19,97%

Source: Elaborated by the Author

4 RESULTS

The results of the experiments are displayed and highlighted in the next paragraph.

4.1 RESULTS WITH RESNET BACKBONE

A total of 9 models with the parameters highlighted in Table 7 shall be of interest for this thesis. They all had the same learning rate and batch size to compare the results better.

Table 7 – Parameters of the Models Training on the ResNet Backbone

		Parameters			
	Name	Size	LeadNr	Batch-Size	Learning rate
1	RES18_arr_1e3_filt_max50_norm_1L_128b	18	1	128	0,001
2	RES18_Arr_1e3_filt_max50_250	18	3	128	0,001
3	RES18_arr_1e3_filt_max50_norm_3L_	18	3	128	0,001
4	RES18_arr_1e3_filt_max50_norm_12L_128b	18	12	128	0,001
5	RES18_arr_1e3_filt_max50_norm_12L_	18	12	128	0,001
6	RES34_arr_1e3_filt_max50_norm_1L_128b	34	1	128	0,001
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	34	3	128	0,001
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	34	3	128	0,001
9	RES34_arr_1e3_filt_max50_norm_12L	34	12	128	0,001

Table 8 showcases the results, with the light green being the best value obtained with size 18 (or 34) and the darker green being the best value overall. The best values obtained with a ResNet18 were 77.8% in validation and 77.9% in the full test accuracy. Almost all of the remaining values also gave the best results for this model when using one lead, but we can see that the overall spread of performance achieved for one and three leads for this model size is fairly small, while the models with 12 leads performed worse. Using the larger model, the best model in validation used 12 leads, achieving a full accuracy of 78.1%, while the best model in the test with three leads achieved a best full accuracy of 78.25%. The difference between the larger and smaller models' performance was small, with some overlap, and the larger models performed slightly better.

Table 8 – Results of the Training on the ResNet Backbone.

	Name	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
		Training					
1	RES18_arr_1e3_filt_max50_norm_1L_128b	0,9624	0,8831	0,8322	0,7987	0,8569	34,415
2	RES18_Arr_1e3_filt_max50_250	0,9601	0,8748	0,8228	0,7826	0,8480	35,980
3	RES18_arr_1e3_filt_max50_norm_3L_	0,9582	0,8686	0,8145	0,7752	0,8407	37,445
4	RES18_arr_1e3_filt_max50_norm_12L_128b	0,9398	0,8297	0,6986	0,6672	0,7585	144,110
5	RES18_arr_1e3_filt_max50_norm_12L_	0,9446	0,8367	0,7347	0,6926	0,7824	135,605
6	RES34_arr_1e3_filt_max50_norm_1L_128b	0,9660	0,8920	0,8517	0,8183	0,8714	31,223
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	0,9574	0,8646	0,8124	0,7707	0,8377	37,723
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	0,9656	0,8905	0,8509	0,8703	0,8400	31,417
9	RES34_arr_1e3_filt_max50_norm_12L	0,9653	0,8918	0,8463	0,8082	0,8684	63,750
		Validation					
1	RES18_arr_1e3_filt_max50_norm_1L_128b	0,9564	0,8684	0,8007	0,7780	0,8332	13,841
2	RES18_Arr_1e3_filt_max50_250	0,9533	0,8575	0,7867	0,7448	0,8206	13,874
3	RES18_arr_1e3_filt_max50_norm_3L_	0,9544	0,8570	0,7975	0,7582	0,8262	13,810
4	RES18_arr_1e3_filt_max50_norm_12L_128b	0,9374	0,8221	0,6882	0,6672	0,7492	20,666
5	RES18_arr_1e3_filt_max50_norm_12L_	0,9454	0,8423	0,7362	0,6996	0,7857	16,400
6	RES34_arr_1e3_filt_max50_norm_1L_128b	0,9543	0,8634	0,8090	0,7714	0,8277	14,303
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	0,9524	0,8549	0,7823	0,7520	0,8170	13,920
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	0,9577	0,8630	0,8183	0,7807	0,8400	13,083
9	RES34_arr_1e3_filt_max50_norm_12L	0,9586	0,8474	0,8221	0,7807	0,8437	28,939
		Test					
1	RES18_arr_1e3_filt_max50_norm_1L_128b	0,9564	0,8682	0,7990	0,7793	0,8321	13,762
2	RES18_Arr_1e3_filt_max50_250	0,9542	0,8575	0,7929	0,7493	0,8239	13,548
3	RES18_arr_1e3_filt_max50_norm_3L_	0,9555	0,8594	0,8018	0,7640	0,8296	13,529
4	RES18_arr_1e3_filt_max50_norm_12L_128b	0,9379	0,8230	0,6891	0,6596	0,7501	20,891
5	RES18_arr_1e3_filt_max50_norm_12L_	0,9452	0,6987	0,7354	0,6987	0,7840	16,310
6	RES34_arr_1e3_filt_max50_norm_1L_128b	0,9550	0,8497	0,8104	0,7771	0,8296	14,210
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	0,9528	0,8547	0,7841	0,7560	0,8179	13,929
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	0,9578	0,8605	0,8213	0,7825	0,8404	13,026
9	RES34_arr_1e3_filt_max50_norm_12L	0,9580	0,8636	0,8190	0,7796	0,8407	13,946

4.2 RESULTS WITH RESNET BACKBONE AND FREQUENCY ANALYSIS

Eleven models with different model sizes, lead numbers, learning rates, and window sizes to make the STFT (50 bins = 200 ms) are of interest here. The exact parameters of the models can be seen in Table 9.

Table 9 – Parameters of the Models Training on the ResNet Backbone with Spectrograms

		Parameters				
	Name	Size	LeadNr	Batch-Size	Learning rate	n_fft
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	18	3	32	0,00100	50
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	18	3	32	0,00100	75
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	34	1	32	0,00100	50
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	34	1	32	0,00100	25
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	34	1	32	0,00100	75
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	34	1	32	0,00100	125
7	RES34_arr_1e3_filt_max50_norm_freq	34	3	32	0,00100	50
8	RES34_arr_1e5_filt_max50_norm_freq	34	3	32	0,00001	50
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	34	3	32	0,00100	75
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	34	3	32	0,00001	75
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	34	3	32	0,00001	125

If we compare the results shared in Table 10, the best model with size 18 achieved a validation full accuracy of 75.86% and in the test 74.17%, performing lower than the equivalent models not using spectrograms. Training the bigger model with spectrograms resulted in the best model achieving 77.75% and 78.45% of full accuracy in validation and test. They were trained with a 0.001 learning rate and a window size of 300 ms. What else we can see from the results with the large models is that some performed similarly to the models with raw data, and some performed worse, not showcasing stable results overall.

Table 10 – Results of the Training on the Resnet Backbone with Spectrograms

	Name	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
		Training					
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	0,9579	0,8809	0,7968	0,7586	0,8367	160,16
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	0,9441	0,8418	0,7228	0,6837	0,7778	209,09
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	0,9341	0,8115	0,6692	0,6386	0,7335	243,17
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	0,9649	0,8964	0,8380	0,8015	0,8662	130,10
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	0,9648	0,8966	0,8364	0,8047	0,8655	132,98
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	0,9542	0,8671	0,7819	0,7508	0,8223	172,11
7	RES34_arr_1e3_filt_max50_norm_freq	0,9585	0,8788	0,8043	0,7714	0,8399	157,42
8	RES34_arr_1e5_filt_max50_norm_freq	0,9538	0,8694	0,7755	0,7358	0,8198	172,35
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	0,9677	0,9061	0,8492	0,8178	0,8767	123,68
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	0,9544	0,8731	0,7765	0,7380	0,8220	169,06
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	0,9728	0,9353	0,8585	0,8305	0,8953	109,90
		Validation					
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	0,9524	0,8679	0,7664	0,7359	0,8140	59,660
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	0,9383	0,8273	0,6894	0,6677	0,7521	76,941
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	0,9352	0,7943	0,7052	0,6495	0,7471	78,867
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	0,9503	0,8333	0,7928	0,7437	0,8126	65,025
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	0,9574	0,8590	0,8216	0,7761	0,8399	55,958
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	0,9524	0,8573	0,7790	0,7491	0,8163	60,427
7	RES34_arr_1e3_filt_max50_norm_freq	0,9423	0,8176	0,7409	0,7032	0,7774	76,210
8	RES34_arr_1e5_filt_max50_norm_freq	0,9470	0,8564	0,7324	0,7032	0,7895	66,189
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	0,9587	0,8747	0,8123	0,7775	0,8424	52,628
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	0,9474	0,8407	0,7562	0,7152	0,7962	65,888
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	0,9474	0,8369	0,7608	0,7079	0,8953	67,877
		Test					
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	0,9532	0,8686	0,7703	0,7417	0,8165	58,652
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	0,9389	0,8289	0,6909	0,6699	0,7536	75,379
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	0,9352	0,7922	0,7059	0,6482	0,7466	78,365
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	0,9511	0,8347	0,7958	0,7476	0,8148	65,160
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	0,9576	0,8575	0,8235	0,7745	0,8401	55,430
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	0,9500	0,8437	0,7737	0,7412	0,8072	63,673
7	RES34_arr_1e3_filt_max50_norm_freq	0,9427	0,8176	0,7419	0,7048	0,7779	75,623
8	RES34_arr_1e5_filt_max50_norm_freq	0,9471	0,8531	0,7354	0,7069	0,7899	66,138
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	0,9600	0,8768	0,8191	0,7848	0,8470	51,215
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	0,9479	0,8404	0,7587	0,7195	0,7975	65,359
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	0,9475	0,8371	0,7600	0,7096	0,7967	67,550

4.3 RESULTS OF EVALUATION OF BACKBONE WITH WINDOWING

The evaluations of the models in Table 11 for the training without using spectrograms show a similar pattern than in the simple test cases, with accuracies coming close to before being more than 1% lower than the test case with the same models having the best full accuracies. For the smaller model, the best accuracies

achieved were 76.41% and 76.34%, while for the best (and bigger) model, the full accuracy of 77.03% and 76.58% were measured.

Table 11 – Results of the Evaluation of Resnet Backbone Models on the Whole Sequence

	Name	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
		Evaluation Constant Label for Whole Sequence					
1	RES18_arr_1e3_filt_max50_norm_1L_128b	0,9526	0,8504	0,7873	0,7641	0,8177	1.788,6
2	RES18_Arr_1e3_filt_max50_250	0,9516	0,8457	0,7840	0,7365	0,8137	1.753,5
3	RES18_arr_1e3_filt_max50_norm_3L_	0,9525	0,8463	0,7919	0,7518	0,8182	1.754,2
4	RES18_arr_1e3_filt_max50_norm_12L_128b	0,9335	0,8012	0,6744	0,6440	0,7324	2.725,2
5	RES18_arr_1e3_filt_max50_norm_12L_	0,9420	0,8229	0,7261	0,6867	0,7715	2.082,5
6	RES34_arr_1e3_filt_max50_norm_1L_128b	0,9509	0,8324	0,7968	0,7606	0,8142	1.830,3
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	0,9503	0,8434	0,7759	0,7454	0,8082	1.770,7
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	0,9549	0,8481	0,8111	0,7703	0,8292	1.673,1
9	RES34_arr_1e3_filt_max50_norm_12L	0,9554	0,8515	0,8105	0,7661	0,8305	1.729,5
		Evaluation Prediction Summed to Label					
1	RES18_arr_1e3_filt_max50_norm_1L_128b	0,9533	0,8419	0,8055	0,7634	0,8233	3.369,2
2	RES18_Arr_1e3_filt_max50_250	0,9518	0,8330	0,8044	0,7345	0,8184	3.314,9
3	RES18_arr_1e3_filt_max50_norm_3L_	0,9529	0,8376	0,8076	0,7492	0,8224	3.345,2
4	RES18_arr_1e3_filt_max50_norm_12L_128b	0,9341	0,7938	0,6910	0,6485	0,7388	5.336,8
5	RES18_arr_1e3_filt_max50_norm_12L_	0,9428	0,8154	0,7446	0,6883	0,7784	4.066,6
6	RES34_arr_1e3_filt_max50_norm_1L_128b	0,9515	0,8220	0,8173	0,7560	0,8197	3.380,3
7	RES34_arr_1e3_filt_max50_norm_128b_1e3	0,9511	0,8342	0,7956	0,7442	0,8145	3.372,5
8	RES34_arr_1e3_filt_max50_norm_3L_128b_	0,9554	0,8376	0,8303	0,7658	0,8340	3.145,6
9	RES34_arr_1e3_filt_max50_norm_12L	0,9560	0,8416	0,8304	0,7638	0,8360	3.251,7

Evaluating the models using spectrograms showed the same pattern as the evaluation with no spectrograms, also achieving a little lower accuracy than in the simple test, with the best smaller model achieving 73.51% and 73.63% and the best ResNet34 achieving 77.63% and 77.53% as can be seen in Table 12.

Table 12 – Results of the Evaluation of Resnet Backbone Models with Spectrograms on the Whole Sequence

	Name	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
		Evaluation Constant Label for Whole Sequence					
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	0,9517	0,8608	0,7657	0,7351	0,8105	1.871,5
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	0,9359	0,8112	0,6839	0,6603	0,7422	2.411,2
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	0,9328	0,7768	0,7041	0,6426	0,7387	2.503,1
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	0,9487	0,8229	0,7898	0,7381	0,8061	2.092,7
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	0,9549	0,8455	0,8149	0,7643	0,8299	1.762,5
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	0,9479	0,8341	0,7665	0,7333	0,7989	2.024,8
7	RES34_arr_1e3_filt_max50_norm_freq	0,9418	0,8111	0,7415	0,7026	0,7747	2.449,8
8	RES34_arr_1e5_filt_max50_norm_freq	0,9449	0,8401	0,7303	0,6996	0,7814	2.113,9
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	0,9579	0,8678	0,8120	0,7763	0,8390	1.636,8
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	0,9455	0,8285	0,7517	0,7093	0,7882	2.093,8
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	0,9458	0,8269	0,7565	0,7045	0,7901	2.170,2
		Evaluation Prediction Summed to Label					
1	RES18_arr_1e3_filt_max50_norm_3L_freq_200ms	0,9519	0,8515	0,7798	0,7363	0,8141	3.661,6
2	RES18_arr_1e3_filt_max50_norm_3L_freq_300ms	0,9365	0,8046	0,6994	0,6666	0,7483	4.745,0
3	RES34_arr_1e3_filt_max50_norm_1L_32b_freq	0,9332	0,7684	0,7235	0,6435	0,7453	4.935,0
4	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_100ms	0,9485	0,8096	0,8085	0,7305	0,8091	3.918,2
5	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_300ms	0,9549	0,8341	0,8313	0,7593	0,8327	3.352,3
6	RES34_arr_1e3_filt_max50_norm_1L_32b_freq_500ms	0,9483	0,8246	0,7837	0,7326	0,8037	3.860,1
7	RES34_arr_1e3_filt_max50_norm_freq	0,9418	0,8015	0,7563	0,6998	0,7782	4.720,0
8	RES34_arr_1e5_filt_max50_norm_freq	0,9456	0,8309	0,7495	0,7030	0,7881	4.110,6
9	RES34_arr_1e3_filt_max50_norm_3L_32b_freq_300ms	0,9583	0,8572	0,8290	0,7735	0,8429	3.101,6
10	RES34_arr_1e5_filt_max50_norm_freq_300ms	0,9456	0,8172	0,7691	0,7079	0,7924	4.049,5
11	RES34_arr_1e5_filt_max50_norm_freq_500ms	0,9457	0,8105	0,7798	0,6987	0,7949	4.077,0

4.4 RESULTS WITH LSTM

In total, 6 LSTM models with different parameters like the base model size or leads, as shown in Table 13, demonstrated some interesting results.

Table 13 – Parameters of the LSTM

		Parameters			
	Name	Size	LeadNr	Batch-Size	Learning rate
1	LSTM18_1L_1e3_arr_filt_norm_fit	18	1	1	0,001
2	LSTM18_3L_1e3_arr_filt_norm_fit	18	3	1	0,001
3	LSTM34_1L_1e3_arr_filt_norm_fit	34	1	1	0,001
4	LSTM34_3L_1e2_arr_filt_norm_fit	34	3	1	0,010
5	LSTM34_3L_1e3_arr_filt_norm_fit_	34	3	1	0,001
6	LSTM34_12L_1e3_arr_filt_norm_fit	34	12	1	0,001

The behaviour we can observe in Table 14 is that the 12-lead larger model performed the best in the training. In contrast, in the test and validation, the smaller model with a ResNet18 backbone with three leads performed for most of the performance calculations considerably better, achieving a full accuracy of 55.97% and 56.91% in validation and test while the best larger model only achieved 43.04% and 43.41%. Furthermore, we can see that the LSTMs were outperformed by the models only using the Resnet backbone.

Table 14 – Results of the LSTM Training

	Name	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
		Training					
1	LSTM18_1L_1e3_arr_filt_norm_fit	0,9442	0,8242	0,7473	0,7228	0,7839	6106,4
2	LSTM18_3L_1e3_arr_filt_norm_fit	0,9431	0,8254	0,7357	0,7133	0,7779	6274,1
3	LSTM34_1L_1e3_arr_filt_norm_fit	0,9448	0,8264	0,7495	0,7239	0,7861	6013,7
4	LSTM34_3L_1e2_arr_filt_norm_fit	0,8353	0,3831	0,3535	0,3502	0,3677	745150,0
5	LSTM34_3L_1e3_arr_filt_norm_fit_	0,9548	0,8526	0,8058	0,7686	0,8286	4778,4
6	LSTM34_12L_1e3_arr_filt_norm_fit	0,9591	0,8697	0,8215	0,7925	0,8449	4435,3
		Validation					
1	LSTM18_1L_1e3_arr_filt_norm_fit	0,8769	0,5634	0,4162	0,3857	0,4788	4582,4
2	LSTM18_3L_1e3_arr_filt_norm_fit	0,9123	0,7277	0,5668	0,5597	0,6373	3258,4
3	LSTM34_1L_1e3_arr_filt_norm_fit	0,8897	0,6469	0,4136	0,3791	0,5046	4705,7
4	LSTM34_3L_1e2_arr_filt_norm_fit	0,8363	0,3884	0,3574	0,3539	0,3723	246962,5
5	LSTM34_3L_1e3_arr_filt_norm_fit_	0,8732	0,5389	0,4602	0,3508	0,4965	4419,2
6	LSTM34_12L_1e3_arr_filt_norm_fit	0,8909	0,5966	0,6078	0,4304	0,6021	4044,0
		Test					
1	LSTM18_1L_1e3_arr_filt_norm_fit	0,8787	0,5696	0,4232	0,3936	0,4856	4518,5
2	LSTM18_3L_1e3_arr_filt_norm_fit	0,9141	0,7315	0,5761	0,5691	0,6446	3172,8
3	LSTM34_1L_1e3_arr_filt_norm_fit	0,8905	0,6501	0,4123	0,3765	0,5046	4670,6
4	LSTM34_3L_1e2_arr_filt_norm_fit	0,8356	0,3832	0,3542	0,3511	0,3682	248037,5
5	LSTM34_3L_1e3_arr_filt_norm_fit_	0,8741	0,5401	0,4636	0,3581	0,4989	4395,0
6	LSTM34_12L_1e3_arr_filt_norm_fit	0,8920	0,5985	0,6113	0,4341	0,6048	4011,4

4.5 RESULTS WITH LSTM AND FREQUENCY ANALYSIS

The training of the three included LSTMs using spectrograms portrayed in Table 15 showed that the accuracy of the trained models was higher in the validation but lower in the test than the equivalent models using raw data. Compared to all the models using raw data, the performance was lower and did not show benefits. The maximum achieved accuracies were only 43% in validation and 32.6% in the test for

the best-performing models. The accuracies in the training were comparable with the other LSTMs using raw data.

Table 15 – Full Results Table of LSTM Training Using Spectrograms

	Parameters					
Name	Size	LeadNr	Batch-Size	Learning rate	n_fft	
LSTM34_1L_1e4_arr_filt_norm_fit_freq	34	1	1	0,0001	50	
LSTM34_3L_1e4_arr_filt_norm_fit_freq_re	34	3	1	0,0001	50	
LSTM34_3L_1e3_arr_filt_norm_fit_freq	34	3	1	0,0010	50	
	Accuracy	Precision	Recall	Accuracy-Full	F1-Score	Loss
	Training					
LSTM34_1L_1e4_arr_filt_norm_fit_freq	0,9379	0,8224	0,6910	0,6805	0,7510	7.315,0
LSTM34_3L_1e4_arr_filt_norm_fit_freq_re	0,9520	0,8492	0,7849	0,7545	0,8158	5.195,4
LSTM34_3L_1e3_arr_filt_norm_fit_freq	0,9529	0,8472	0,7960	0,7613	0,8208	5.026,7
	Validation					
LSTM34_1L_1e4_arr_filt_norm_fit_freq	0,8484	0,3873	0,1995	0,1587	0,2633	6.419,5
LSTM34_3L_1e4_arr_filt_norm_fit_freq_re	0,8625	0,4924	0,3816	0,4300	0,2796	4.912,9
LSTM34_3L_1e3_arr_filt_norm_fit_freq	0,8721	0,5341	0,4582	0,3258	0,4932	4.718,9
	Test					
LSTM34_1L_1e4_arr_filt_norm_fit_freq	0,8492	0,3881	0,2001	0,1605	0,2641	6.404,9
LSTM34_3L_1e4_arr_filt_norm_fit_freq_re	0,8631	0,4920	0,3807	0,2807	0,4293	4.897,3
LSTM34_3L_1e3_arr_filt_norm_fit_freq	0,8723	0,5323	0,4609	0,3260	0,4940	4.727,7

5 DISCUSSION

The results of the research and experiments will be discussed in the following.

5.1 EFFECT OF THE NUMBER OF LEADS

As shown by the results, there were some differences in the accuracies of models using different numbers of leads. Those differences were often small, especially between using three and twelve leads. It can be explained by the chosen set of detectable arrhythmias, especially in leads II, V1 and V5, which professionals commonly use to classify atrial arrhythmias. Using only lead II and the three leads, as just mentioned, we can expect more valuable information using three leads, but using all 12 leads will not give a significantly higher amount of information. Interestingly, the ResNet18 with only one lead performed better than using multiple leads, maybe because of the increased complexity of processing the information of multiple leads.

One has to know one's data and its arrhythmias as a specific lead combination and specific leads are more important and can reduce the dimension of input data while achieving similar results. Therefore, adjusting the number and types of leads used for the problem is very important, as some leads do not have extra information.

5.2 EFFECT OF THE MODEL SIZE

The model size of the classifiers had different effects. In the simple models, we did only achieve minor gains in accuracy. One possible explanation is that we already have a good representation with the ResNet18, which is, in fact, already quite a powerful network structure. Therefore, a bigger network did not give us a significantly better representation. Suppose we wanted to increase the performance with a bigger network. In that case, we might have to consider increasing the training data and model size to counteract the possibility of an overfit on the data or use more complicated features, which at the same time also need to transmit more information better representing the arrhythmias. Another option to increase the training data and its variability is using augmentation techniques specific to ECGs, as data augmentation is a very common and inexpensive way to do so. However, one must be mindful only to implement methods guaranteeing the label invariability with augmentation. It can further be expected to have a bigger difference in accuracy if we increase the number

of labels because the network will have to learn more different patterns and representations.

Using the LSTMs with a different backbone size seemed to have the opposite effect of what should be expected because the smaller backbones performed better in the evaluations. The most likely explanation is that it overfitted less with the smaller backbone, while overfitting was one of the general problems, as will be discussed in the next chapter.

5.3 EFFECTS WHEN USING A RECURRENT NEURAL NETWORK

Using an LSTM did not have the desired improvement in the accuracies of the classifiers. On average, the accuracies were significantly lower than just training using the backbone with a very limited length of 5 seconds.

One suggested explanation is that a large part of the training samples was short, so the chance of the arrhythmias being in the first seconds is fairly high. If the arrhythmias already occur in the first seconds, it makes the extra summing up of features over time obsolete, and therefore, we cannot expect an improvement using an LSTM. If our training data was, on average, created with long samples with only very spread-out arrhythmias, using only the base without LSTM for training would likely not yield good results, making using a classifier with an LSTM more reliable.

Another thing we can observe in the accuracies is that if we compare the full training accuracy with the validation and test accuracy, the training accuracy is much higher. This suggests that the model was overfitted, which could be explained by the LSTM being a bigger, more complicated structure with more weight. We could try to counteract that by using a smaller base, like a smaller, simpler CNN, to calculate the features or use fewer features in the hidden and cell state of the LSTM.

Moreover, it must be stated that the training of the utilized LSTM is very difficult compared to the base ResNet. First, one problem is that only a batch size of one can be used, as the samples have different lengths. This slows down the training considerably. It was tried in this thesis to pad all the samples to the same length, but as the sample size differs a lot, it is inefficient in terms of memory. Another possibility is to divide the set into sets of different lengths covering a certain range, where we pad the samples to the maximum range length and use different batch sizes for the different ranges. This is effective, but training a model in this fashion and padding the samples

also yielded bad results. An explanation is that one must be careful how to pad, as simply padding to 0 or a mean might give bad gradients for the training of a longer LSTM. Therefore, training with batch size 1 had to be practiced.

Another idea for improving the training of this approach would be giving an output of each LSTM cell and comparing it to the labels, as it will give additional gradients for each timestep and not one that has to be passed down the complete time series. This was not done because we only have one label vector for each sample with no time information. Therefore, giving labels to a cell might also negatively affect it, where the arrhythmias might not yet have occurred.

Using a pre-trained base, which creates the feature vectors for the training, would be another option. It would result in a very good comparison of using an LSTM with the base instead of only using the base, as the difference in performance would only be dependent on the LSTM and would ensure a good base for its training.

Another topic needing to be discussed here is that a very simple LSTM with depth one and 512 features was used. Therefore, we could also try to use even fewer features to counteract overfitting and simplify the model or make the model more complicated, trying to process the feature vectors better. Options for that would be to make the model deeper or use a bi-directional LSTM, where we can see in a very similar approach by Chen et al. (Chen et al. 2019) a model performing well.

Finally, we must mention the big advantage of an LSTM, which unfortunately did not achieve the desired results in this work. This advantage would be the ability, to sum up a whole time sequence very effectively and, even more importantly – it can be adapted to giving real-time outputs while considering the past inputs, which should stabilize and improve the results of a real-world used classifier.

5.4 EFFECT OF SPECTROGRAMS WITH DIFFERENT WINDOW SIZES COMPARED TO RAW SIGNALS

Using spectrograms as features instead of using the raw signal did give mixed results; some of the models seemed to perform not as well as those using raw signals, but the best models using spectrograms did perform as well or even a little better than using raw signals.

This can be explained by the fact that even though the spectrogram brings further information, the extra frequency information was unnecessary for the task, not

giving advantages. Another possibility is that it might even have hindered the training and that for ECGs, the time resolution being worse in a spectrogram could not be counteracted by the extra frequency information.

When analyzing the results achieved with different window sizes, we can see that of all the sizes, 300 ms seemed to have given most of the time, but with other window sizes, good results were achieved as well as having a non-convex behaviour because a window of 100 ms performed better than 200 ms, and 500 ms performed worse than 300 ms. Depending on the sizes, an advantageous tradeoff of time and frequency resolution is reached; a thorough search could give clarity and an optimized size.

A further hyperparameter one could think about training is the hop size. It was chosen very low to show detailed frequency changes; however, it might not be necessary, and one could reduce the input data size by reducing this parameter if performance stays the same.

A very interesting consideration would be to train a future model using spectrograms and raw signals while combining both advantages to achieve an even better model.

5.5 EVALUATION OF THE RESNET BACKBONES FOR THE WHOLE SAMPLE

In our study, we could show that the loss of accuracy in evaluating the models on the full sample with a moving-window approach as described was minor. Therefore, we can assume that even though we only trained with the first five seconds, we still got a good representation. This is most likely because many samples were very short, so the arrhythmias would likely occur in the first five seconds. A dataset with long samples and spread arrhythmias would be interesting to verify if the approach is still working. Another interesting dataset to work with would be one having time-dependent labels so that a moving window could be trained and then verified more precisely.

5.6 TRAINING WITH FREE RESOURCES

The limits given by the available resources, only having access to the free version of Google Colab as well as to one Laptop with an NVIDIA GFORCE RTX Titan

GPU with 4 GB RAM, give a great insight into the practicality of the training as well as the actual usage of similar models in practice.

The evaluation and test of the classifiers in practice on a computer (especially with graphics card) can be achieved quickly without problem; if many samples need to be analyzed for a study, however, the time needed will increase, needing to be taken into consideration or using a more powerful GPU.

Training models on the used Laptop had its difficulties, as big model sizes and or bigger batch sizes will not fit on the RAM of the GPU. Moreover, the training did bind a large amount of the processing power, CPU, and CPU RAM, making working on projects during training more challenging.

Working with Google Colab was an overall positive experience, as it opened much computational power with bigger GPUs while outsourcing the calculations, not slowing down the user's device. Without Colab, achieving the results for this thesis would have been a lot more challenging.

However, one should not forget that the time limits are strict, only being able to, on average, train around 4 hours every 24 hours while needing to be connected to the internet and active in the Colab notebook for the training to continue. Furthermore, resources are not reliably available, as it can (and did) happen that all the resources are occupied by paid accounts or Google itself, which can even happen for several days.

The limited amount of resources also did not allow the retraining of models on a larger scale to have more statistical validation of the performance of a model. Furthermore, it slowed hyperparameter search, such as learning rates and or window sizes and limited the experiments possible in the time given, especially for the more complex models training longer.

5.7 TRAINING TIMES

The training times of the models showed an expected pattern. The very small ResNet18-based models train the fastest, and the ResNet34-based models longer. More leads will augment the time needed, and more data must be processed. Utilizing spectrograms will also increase the training times, especially when the ready spectrograms are not yet saved in the cache needing to make the STFT. It is very important to cache the data as it did speed up the training in many cases by multiples.

Training LSTMs increased training times even more due to the small batch size used because the samples were different lengths. To know the times, the ResNet-based models could normally be trained on Colab within one to a few days of training (with the limitations of using the free Resources of Colab), while the bigger LSTMs could take weeks to train.

Concerning the training times compared to the achieved results, one must conclude from this thesis that the complex models with the LSTMs did not pay off concerning the training times, as their performance, as with reasons discussed above, was lower while their training times were much higher. Using a simpler model, mostly when having limited resources, is a consideration one should take, as they are, in many cases, sufficient, having only minor differences in performance while being simpler and with fewer resources to handle and faster to train.

5.8 EVALUATION OF THE RESULTS OF THIS WORK TO REAL-WORLD APPLICATIONS

Using the models trained here for real-world applications would be possible quite easily. The only necessity would be creating an interface to transform the digitized ECGs and other essential data, like the sampling rate, into a format the algorithm can read. Furthermore, the output must be transformed into an easily usable format. All this would have to be done in an easy executable and installable program with all the necessary trained models integrated, which could be installed on any hospital computer, preferably one with more computational power and a graphic card. Another option would be to install the program on a server so that this application could run remotely online, which would not require the user to have a good computer.

To the question of where to use an algorithm like the one created here, two possible ways seem to be the most reasonable. First, we could use it as an advice tool for the medical environment. As the classifications are imperfect so far, diagnosing based only on the algorithm is not advisable. However, the algorithm still can be used as a safety measure, which can register arrhythmias that would have otherwise been overseen, being very important, for example, in the constant supervision of a patient. The classification can also allow the cardiologist to diagnose early on to make diagnosing faster and improve the workflow in, e.g., a hospital.

A second possible usage is for research. If we have large amounts of data needing analysis, e.g., a study, getting qualified personnel to label is time-consuming and cost-intensive. If we know the error in our classification system, we can analyze those large amounts of data rapidly and automatically while still having a statistically meaningful result within our error rate. In many cases, this can help derive a new and important insight into our dataset.

5.9 LIMITATIONS OF THIS THESIS

One limitation is the very restricted resources available for training the models. This reduced the possibility of retraining the models, which would help to get different initializations and random processes during the training for a more complete picture of the models. Furthermore, it limited the number of parameters and models and tried to get better comparisons. On top of that, it limited the hyperparameter optimization, especially in the models with a lot, e.g., when using spectrograms, the models might not be optimized to the best available ones. This was especially an issue when the models had very long training times, like when training with the underlying LSTMs. Therefore, we only get a limited view of the possible performance of the models.

Another limitation of this thesis is given by the data available. Even though we have very good and large datasets from various sources, this does not guarantee that all more specific environments for measuring ECGs are accounted for. This can include regional differences in making ECGs, uncommon in countries where the data is from. Moreover, the labels of the datasets will most likely have some errors and be partly unbalanced. This will also affect the training of the models influencing performance. Therefore, at least testing the models on some different unseen data might give some more insights about the true performance of the in this thesis tested structures.

Lastly, the models were never used in a field test; therefore, we cannot make any certified observations on their performance and utility in a real environment. Due to that, we can only make assumptions about their performance based on the experiments, limiting the expressiveness regarding that. That, of course, could mean that the model could perform worse in an application. Therefore, we will have to test the model in a more real environment for verification before it can be applied on a larger scale.

6 CONCLUSION AND OUTLOOK

In the following, the whole work will be briefly summarized to conclude then. Future work will be outlined as well.

As cardiac arrhythmias are a very common concern in healthcare, it is important to be able to diagnose and treat them. ECGs are a very common tool to gain insights into the beating process of the heart, as the heart emits an electric signal during its different stations while beating. This signal can be measured at different points, giving different insights and creating various channels. As deep learning advances, it is possible to create automated classification systems, as is done in this thesis. Creating a training database is very time-consuming and labor-consuming; for this thesis, the 2021 Physionet challenge data was used, as this dataset provides an ample dataset for classification.

For this thesis, several state-of-the-art methods were tested. Simple models based on ResNet18 and ResNet34 being trained on the first five seconds for various lead combinations were tried. As input data, either the raw data was slightly modified by filtering and normalization, or spectrograms were created using different window sizes. These models were also evaluated with a moving window approach on the whole sample. Furthermore, those models were utilized as backbones to create feature vectors fed into a simple one-layer/one-directional LSTM to analyze and train on samples with different lengths.

The training and testing of the models resulted in the simple models only analyzing five seconds performing well, while the LSTMs did not meet the expectations.

Based on the results of the discussions, it can be concluded that using all leads is unnecessary, as 12 leads did not improve performance, as the best models were only trained using three leads. Furthermore, only little improvement in the accuracies of the best models using spectrograms was found, while the average using spectrograms had lower performance, resulting in the conclusion that the creation of spectrograms did not have great advantages. The model size using ResNet18 and ResNet34 based models showed little more performance for the bigger model, concluding that utilizing a bigger model is the better choice, however only by little difference. Using LSTMs, this was the opposite, as the LSTMs with ResNet18

performed better, most likely due to overfitting. Using LSTMs did not achieve the opposite result of what was expected as they did not perform well, concluding that it is sometimes preferable to use simpler network structures. This is also true because while evaluating the five-second models on the whole sample, good results were achieved, outperforming the LSTMs proving their value and usability. From the resources available, the observation was made that simple models are also favorable due to simpler training and faster training times and being able to do more experiments with very limited resources. Furthermore, possible usage of the models was outlined in different settings that were easily applicable without much extra work.

In total, it can be concluded that it was possible to create a simple classifier for atrial arrhythmias with good results and fast training times in the form of a ResNet34 classifier either using raw data or spectrograms as can be seen by the accuracies achieved in test and evaluation. Furthermore, the conclusion can be drawn that extending the structure by an LSTM did not improve performance.

Investigations driving this project further on developing the classifier could include a more thorough hyperparameter search for creating the spectrograms to achieve better performance or creating a classifier integrating spectrograms and raw data to use both. On top of that, a more optimized approach for optimizing a moving window for the classification could be tested. Another hyperparameter search could also be conducted to train LSTMs to find better values to improve their performance. On top of that, simpler neural networks to create the input could be counteracting overfitting more efficiently or using pre-initialized networks to create the features for performance. If other databases are available, validating their results would also be interesting. Validating the classifier for each class would also give insights into problems classifying some of the chosen arrhythmias. Furthermore, more classifiers trained to find other diseases could be trained with the same pipeline.

Moreover, a common interface to transmit the data must be defined for the network to be usable in real-world cases. After that, a field test could be made to show the usability and applicability and give valuable insights into where improvements must be made.

REFERENCES

Virani SS, Alonso A, Aparicio HJ, Benjamin EJ, Bittencourt MS, Callaway CW, et al. Heart Disease and Stroke Statistics – 2021 Update: a Report from the American Heart Association. Circulation 2021;143(8):e254–e743

Sattar Y, Chhabra L. Electrocardiogram. 2023 Apr 3. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan–. PMID: 31747210.

Kligfield, Paul. (2002). The centennial of the Einthoven electrocardiogram. Journal of electrocardiology. 35 Suppl. 123-9. 10.1054/jelc.2002.37169.

Aldrich HR, Hindman NB, Hinohara T, Jones MG, Boswick J, Lee KL, et al. Identification of the Optimal Electrocardiographic Leads for Detecting Acute Epicardial Injury in Acute Myocardial Infarction. The American Journal of Cardiology 1987;59(1):20–23.

Drew BJ, Pelter MM, Brodnick DE, Yadav AV, Dempel D, Adams MG. Comparison of a New Reduced Lead Set ECG with the Standard ECG for Diagnosing Cardiac Arrhythmias and Myocardial Ischemia. Journal of Electrocardiology 2002;35(4, Part B):13–21.

Green M, Ohlsson M, Lundager Forberg J, Björk J, Edenbrandt L, Ekelund U. Best Leads in the Standard Electrocardiogram for the Emergency Detection of Acute Coronary Syndrome. Journal of Electrocardiology 2007;40(3):251–256.

Reyna, M., Sadr, N., Perez Alday, E. A., Liu, C., Seyedi, S., & Clifford, G. D. (2020). Will Two Do? Varying Dimensions in Electrocardiography: the PhysioNet - Computing in Cardiology Challenge 2021 (version 1.0.0). PhysioNet. https://doi.org/10.13026/g9wd-3j66.

PhysioNet/Computing in Cardiology Challenge 2021. https://physionetchallenges.org/2021/. Acesso no dia 30/09/2021.

Perez Alday EA, Gu A, Shah AJ, Robichaux C, Wong AI, Liu C, Liu F, Rad AB, Elola A, Seyedi S, Li Q, Sharma A, Clifford GD, Reyna MA. Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020. Physiol Meas. 2020 Nov 11. http://doi.org/10.1088/1361-6579/abc960.

Zhu, Zhaowei & Lan, Xiang & Zhao, Tingting & Guo, Yangming & Kojodjojo, Pipin & Xu, Zhuoyang & Liu, Zhuo & Liu, Siqi & Wang, Han & Sun, Xingzhi & Feng, Mengling. (2021). Identification of 27 abnormalities from multi-lead ECG signals: an ensembled SE_ResNet framework with Sign Loss function. Physiological Measurement. 42. 10.1088/1361-6579/ac08e6.

Singstad, Bjørn-Jostein & Tronstad, Christian. (2020). Convolutional Neural Network and Rule-Based Algorithms for Classifying 12-lead ECGs. 10.22489/CinC.2020.227.

Mostayed, Ahmed & Luo, Junye & Shu, Xingliang & Wee, William. (2018). Classification of 12-Lead ECG Signals with Bi-directional LSTM Network.

Wong, Alexander & Salimi, Amir & Hindle, Abram & Kalmady, Sunil & Kaul, Padma. (2021). Multilabel 12-Lead Electrocardiogram Classification Using Beat to Sequence Autoencoders. 1270-1274. 10.1109/ICASSP39728.2021.9414934.

Xu, Xue & Jeong, Sohyun & Li, Jianqiang. (2020). Interpretation of Electrocardiogram (ECG) Rhythm by Combined CNN and biLSTM. IEEE Access. PP. 1-1. 10.1109/ACCESS.2020.3006707.

Chen, Yu-Jhen & Liu, Chien-Liang & Tseng, Vincent & Hu, Yu-Feng & Chen, Shih-Ann. (2019). Large-scale Classification of 12-lead ECG with Deep Learning. 1-4. 10.1109/BHI.2019.8834468.

Natarajan, Annamalai & Chang, Yale & Mariani, Sara & Rahman, Asif & Boverman, Greg & Vij, Shruti & Rubin, Jonathan. (2020). A Wide and Deep Transformer Neural Network for 12-Lead ECG Classification. 10.22489/CinC.2020.107.

Arpitha, Yalamanchili & Madhumathi, G. & Balaji, N.. (2021). Spectrogram analysis of ECG signal and classification efficiency using MFCC feature extraction technique. Journal of Ambient Intelligence and Humanized Computing. 10.1007/s12652-021-02926-2.

Huang, Jingshan & Chen, Binqiang & Yao, Bin & He, Wp. (2019). ECG Arrhythmia Classification Using STFT-Based Spectrogram and Convolutional Neural Network. IEEE Access. PP. 1-1. 10.1109/ACCESS.2019.2928017.

Gupta, Varun & Mittal, Monika & Mittal, Vikas & Gupta, Anshu. (2021). ECG signal analysis using CWT, spectrogram and autoregressive technique. Iran Journal of Computer Science. 10.1007/s42044-021-00080-8.

Ignacio, Paul Samuel & Bulauan, Jay-Anne & Manzanares, John Rick. (2020). A Topology Informed Random Forest Classifier for ECG Classification. 10.22489/CinC.2020.297.

Sambit Mahapatra. (2018, March 21). Why Deep Learning over Traditional Machine Learning?. Towards Data Science. Access link: https://towardsdatascience.com/why-deep-learning-is-needed-over-traditional-machine-learning-1b6a99177063

Nonaka, Naoki & Seita, Jun. (2020). Data Augmentation for Electrocardiogram Classification with Deep Neural Network.

Romdhane TF, Alhichri H Dr, Ouni R Dr, Atri M Pr. Electrocardiogram heartbeat classification based on a deep convolutional neural network and focal loss. Comput Biol Med. 2020 Aug;123:103866. doi: 10.1016/j.compbiomed.2020.103866. Epub 2020 Jul 5. PMID: 32658786.

Hsieh CH, Li YS, Hwang BJ, Hsiao CH. Detection of Atrial Fibrillation Using 1D Convolutional Neural Network. Sensors (Basel). 2020 Apr 10;20(7):2136. doi: 10.3390/s20072136. PMID: 32290113; PMCID: PMC7180882.

Tutuko B, Nurmaini S, Tondas AE, Rachmatullah MN, Darmawahyuni A, Esafri R, Firdaus F, Sapitri AI. AFibNet: an implementation of atrial fibrillation detection with

convolutional neural network. BMC Med Inform Decis Mak. 2021 Jul 14;21(1):216. doi: 10.1186/s12911-021-01571-1. PMID: 34261486; PMCID: PMC8281594.

Wu Q, Sun Y, Yan H, Wu X. ECG signal classification with binarized convolutional neural network. Comput Biol Med. 2020 Jun;121:103800. doi: 10.1016/j.compbiomed.2020.103800. Epub 2020 May 5. PMID: 32568678.

Antoni Ľ, Bruoth E, Bugata P, Bugata P Jr, Gajdoš D, Horvát Š, Hudák D, Kmečová V, Staňa R, Staňková M, Szabari A, Vozáriková G. Automatic ECG classification and label quality in training data. Physiol Meas. 2022 Jun 28;43(6). doi: 10.1088/1361-6579/ac69a8. PMID: 35453131.

Krasteva V, Christov I, Naydenov S, Stoyanov T, Jekova I. Application of Dense Neural Networks for Detection of Atrial Fibrillation and Ranking of Augmented ECG Feature Set. Sensors (Basel). 2021 Oct 15;21(20):6848. doi: 10.3390/s21206848. PMID: 34696061; PMCID: PMC8538849.

Zhang H, Liu C, Zhang Z, Xing Y, Liu X, Dong R, He Y, Xia L, Liu F. Recurrence Plot-Based Approach for Cardiac Arrhythmia Classification Using Inception-ResNet-v2. Front Physiol. 2021 May 17;12:648950. doi: 10.3389/fphys.2021.648950. PMID: 34079470; PMCID: PMC8165394.

Alsaleem MN, Islam MS, Al-Ahmadi S, Soudani A. Multiscale Encoding of Electrocardiogram Signals with a Residual Network for the Detection of Atrial Fibrillation. Bioengineering (Basel). 2022 Sep 16;9(9):480. doi: 10.3390/bioengineering9090480. PMID: 36135025; PMCID: PMC9495512.

He Z, Yuan Z, An P, Zhao J, Du B. MFB-LANN: A lightweight and updatable myocardial infarction diagnosis system based on convolutional neural networks and active learning. Comput Methods Programs Biomed. 2021 Oct;210:106379. doi: 10.1016/j.cmpb.2021.106379. Epub 2021 Aug 28. PMID: 34517182.

Lee H, Shin M. Learning Explainable Time-Morphology Patterns for Automatic Arrhythmia Classification from Short Single-Lead ECGs. Sensors (Basel). 2021 Jun 24;21(13):4331. doi: 10.3390/s21134331. PMID: 34202805; PMCID: PMC8272104.

Nasim A, Kim YS. DE-PNN: Differential Evolution-Based Feature Optimization with Probabilistic Neural Network for Imbalanced Arrhythmia Classification. Sensors (Basel). 2022 Jun 12;22(12):4450. doi: 10.3390/s22124450. PMID: 35746232; PMCID: PMC9227752.

Sbrollini A, Tomassini S, Emaldi E, Marcantoni I, Morettini M, Dragoni AF, Burattini L. Multiclass Convolutional Neural Networks for Atrial Fibrillation Classification. Annu Int Conf IEEE Eng Med Biol Soc. 2022 Jul;2022:1288-1291. doi: 10.1109/EMBC48229.2022.9871124. PMID: 36086141.

Jekova I, Christov I, Krasteva V. Atrioventricular Synchronization for Detection of Atrial Fibrillation and Flutter in One to Twelve ECG Leads Using a Dense Neural Network Classifier. Sensors (Basel). 2022 Aug 14;22(16):6071. doi: 10.3390/s22166071. PMID: 36015834; PMCID: PMC9413391.

Jiang M, Lu Y, Li Y, Xiang Y, Zhang J, Wang Z. [Research on electrocardiogram classification using deep residual network with pyramid convolution structure]. Sheng Wu Yi Xue Gong Cheng Xue Za Zhi. 2020 Aug 25;37(4):692-698. Chinese. doi: 10.7507/1001-5515.201912048. PMID: 32840087.

Prabhakararao E, Dandapat S. Atrial Fibrillation Burden Estimation using Multi-Task Deep Convolutional Neural Network. IEEE J Biomed Health Inform. 2022 Jul 18; PP. doi: 10.1109/JBHI.2022.3191682. Epub ahead of print. PMID: 35849681.

Aublin PG, Ben Ammar M, Fix J, Barret M, Behar JA, Oster J. Predict alone, decide together: cardiac abnormality detection based on single lead classifier voting. Physiol Meas. 2022 May 25;43(5). doi: 10.1088/1361-6579/ac66b9. PMID: 35413703.

Srivastava A, Pratiher S, Alam S, Hari A, Banerjee N, Ghosh N, Patra A. A deep residual inception network with channel attention modules for multi-label cardiac abnormality detection from reduced-lead ECG. Physiol Meas. 2022 Jun 28;43(6). doi: 10.1088/1361-6579/ac6f40. PMID: 35550571.

Rubin J, Parvaneh S, Rahman A, Conroy B, Babaeizadeh S. Densely connected convolutional networks for detection of atrial fibrillation from short single-lead ECG recordings. J Electrocardiol. 2018 Nov-Dec;51(6S):S18-S21. doi: 10.1016/j.jelectrocard.2018.08.008. Epub 2018 Aug 10. PMID: 30122456.

Ge R, Shen T, Zhou Y, Liu C, Zhang L, Yang B, Yan Y, Coatrieux JL, Chen Y. Convolutional squeeze-and-excitation network for ECG arrhythmia detection. Artif Intell Med. 2021 Nov; 121:102181. doi: 10.1016/j.artmed.2021.102181. Epub 2021 Sep 25. PMID: 34763803.

Yang J, Cai W, Wang M. Premature beats detection based on a novel convolutional neural network. Physiol Meas. 2021 Jul 28;42(7). doi: 10.1088/1361-6579/ac0e82. PMID: 34167103.

Zhu Z, Lan X, Zhao T, Guo Y, Kojodjojo P, Xu Z, Liu Z, Liu S, Wang H, Sun X, Feng M. Identification of 27 abnormalities from multi-lead ECG signals: an ensembled SE_ResNet framework with Sign Loss function. Physiol Meas. 2021 Jun 29;42(6). doi: 10.1088/1361-6579/ac08e6. PMID: 34098532.

Xu Z, Guo Y, Zhao T, Zhao Y, Liu Z, Sun X, Xie G, Li Y. Abnormality classification from electrocardiograms with various lead combinations. Physiol Meas. 2022 Jul 18;43(7). doi: 10.1088/1361-6579/ac70a4. PMID: 35580597.

Zhao Z, Murphy D, Gifford H, Williams S, Darlington A, Relton SD, Fang H, Wong DC. Analysis of an adaptive lead weighted ResNet for multiclass classification of 12-lead ECGs. Physiol Meas. 2022 Apr 4;43(3). doi: 10.1088/1361-6579/ac5b4a. PMID: 35255483.

Sawant NK, Patidar S. Application of Fourier-Bessel expansion and LSTM on multi-lead ECG for cardiac abnormalities identification. Physiol Meas. 2022 Nov 21. doi: 10.1088/1361-6579/aca4b9. Epub ahead of print. PMID: 36410043.

Cheng J, Zou Q, Zhao Y. ECG signal classification based on deep CNN and BiLSTM. BMC Med Inform Decis Mak. 2021 Dec 28;21(1):365. doi: 10.1186/s12911-021-01736-y. PMID: 34963455; PMCID: PMC8715576.

Wang J. Automated detection of premature ventricular contraction based on the improved gated recurrent unit network. Comput Methods Programs Biomed. 2021 Sep;208:106284. doi: 10.1016/j.cmpb.2021.106284. Epub 2021 Jul 16. PMID: 34304005.

Xiong Z, Nash MP, Cheng E, Fedorov VV, Stiles MK, Zhao J. ECG signal classification for the detection of cardiac arrhythmias using a convolutional recurrent neural network. Physiol Meas. 2018 Sep 24;39(9):094006. doi: 10.1088/1361-6579/aad9ed. PMID: 30102248; PMCID: PMC6377428.

Ping Y, Chen C, Wu L, Wang Y, Shu M. Automatic Detection of Atrial Fibrillation Based on CNN-LSTM and Shortcut Connection. Healthcare (Basel). 2020 May 20;8(2):139. doi: 10.3390/healthcare8020139. PMID: 32443926; PMCID: PMC7348856.

Liang Y, Yin S, Tang Q, Zheng Z, Elgendi M, Chen Z. Deep Learning Algorithm Classifies Heartbeat Events Based on Electrocardiogram Signals. Front Physiol. 2020 Oct 2;11:569050. doi: 10.3389/fphys.2020.569050. PMID: 33117191; PMCID: PMC7566908.

Yildirim O, Talo M, Ciaccio EJ, Tan RS, Acharya UR. Accurate deep neural network model to detect cardiac arrhythmia on more than 10,000 individual subject ECG records. Comput Methods Programs Biomed. 2020 Dec;197:105740. doi: 10.1016/j.cmpb.2020.105740. Epub 2020 Sep 8. PMID: 32932129; PMCID: PMC7477611.

Kang J, Wen H. A study on several critical problems on arrhythmia detection using varying-dimensional electrocardiography. Physiol Meas. 2022 Jun 28;43(6). doi: 10.1088/1361-6579/ac6aa3. PMID: 35472848.

Jiang M, Gu J, Li Y, Wei B, Zhang J, Wang Z, Xia L. HADLN: Hybrid Attention-Based Deep Learning Network for Automated Arrhythmia Classification. Front Physiol. 2021 Jul 5;12:683025. doi: 10.3389/fphys.2021.683025. PMID: 34290619; PMCID: PMC8289344.

Gao Y, Wang H, Liu Z. A Novel Approach for Atrial Fibrillation Signal Identification Based on Temporal Attention Mechanism. Annu Int Conf IEEE Eng Med Biol Soc. 2020 Jul;2020:316-319. doi: 10.1109/EMBC44109.2020.9175823. PMID: 33017992.

Xu P, Liu H, Xie X, Zhou S, Shu M, Wang Y. Interpatient ECG Arrhythmia Detection by Residual Attention CNN. Comput Math Methods Med. 2022 Apr 8;2022:2323625. doi: 10.1155/2022/2323625. PMID: 35432590; PMCID: PMC9012615.

Khamis H, Chen J, Stephen Redmond J, Lovell NH. Detection of Atrial Fibrillation from RR Intervals and PQRST Morphology using a Neural Network Ensemble. Annu Int Conf IEEE Eng Med Biol Soc. 2018 Jul;2018:5998-6001. doi: 10.1109/EMBC.2018.8513496. PMID: 30441703.

Warrick PA, Nabhan Homsi M. Ensembling convolutional and long short-term memory networks for electrocardiogram arrhythmia detection. Physiol Meas. 2018 Oct 30;39(11):114002. doi: 10.1088/1361-6579/aad386. PMID: 30010088.

Rizwan M, Whitaker BM, Anderson DV. AF detection from ECG recordings using feature selection, sparse coding, and ensemble learning. Physiol Meas. 2018 Dec 24;39(12):124007. doi: 10.1088/1361-6579/aaf35b. PMID: 30524091.

Shao M, Bin G, Wu S, Bin G, Huang J, Zhou Z. Detection of atrial fibrillation from ECG recordings using decision tree ensemble with multi-level features. Physiol Meas. 2018 Sep 27;39(9):094008. doi: 10.1088/1361-6579/aadf48. PMID: 30187894.

Raj S, Ray KC. Automated recognition of cardiac arrhythmias using sparse decomposition over composite dictionary. Comput Methods Programs Biomed. 2018 Oct;165:175-186. doi: 10.1016/j.cmpb.2018.08.008. Epub 2018 Aug 22. PMID: 30337072.

Shi J, Chen C, Liu H, Wang Y, Shu M, Zhu Q. Automated Atrial Fibrillation Detection Based on Feature Fusion Using Discriminant Canonical Correlation Analysis. Comput Math Methods Med. 2021 Apr 8;2021:6691177. doi: 10.1155/2021/6691177. PMID: 33897806; PMCID: PMC8052181.

Mukherjee A, Dutta Choudhury A, Datta S, Puri C, Banerjee R, Singh R, Ukil A, Bandyopadhyay S, Pal A, Khandelwal S. Detection of atrial fibrillation and other abnormal rhythms from ECG using a multi-layer classifier architecture. Physiol Meas. 2019 Jun 4;40(5):054006. doi: 10.1088/1361-6579/aaff04. PMID: 30650387.

Zhang J, Liu A, Liang D, Chen X, Gao M. Interpatient ECG Heartbeat Classification with an Adversarial Convolutional Neural Network. J Healthc Eng. 2021 May 29;2021:9946596. doi: 10.1155/2021/9946596. PMID: 34194685; PMCID: PMC8181174.

Sharma M, Singh S, Kumar A, San Tan R, Acharya UR. Automated detection of shockable and non-shockable arrhythmia using novel wavelet-based ECG features. Comput Biol Med. 2019 Dec;115:103446. doi: 10.1016/j.compbiomed.2019.103446. Epub 2019 Sep 18. PMID: 31627019.

Toma TI, Choi S. A Parallel Cross Convolutional Recurrent Neural Network for Automatic Imbalanced ECG Arrhythmia Detection with Continuous Wavelet Transform. Sensors (Basel). 2022 Sep 28;22(19):7396. doi: 10.3390/s22197396. PMID: 36236496; PMCID: PMC9573388.

Bortolan G, Christov I, Simova I. Potential of Rule-Based Methods and Deep Learning Architectures for ECG Diagnostics. Diagnostics (Basel). 2021;11(9):1678. Published 2021 Sep 14. doi:10.3390/diagnostics11091678

Kalidas V, Tamil LS. Detection of atrial fibrillation using discrete-state Markov models and Random Forests. Comput Biol Med. 2019 Oct;113:103386. doi: 10.1016/j.compbiomed.2019.103386. Epub 2019 Aug 8. PMID: 31446318.

Hernandez F, Mendez D, Amado L, Altuve M. Atrial Fibrillation Detection in Short Single Lead ECG Recordings Using Wavelet Transform and Artificial Neural

Networks. Annu Int Conf IEEE Eng Med Biol Soc. 2018 Jul;2018:5982-5985. doi: 10.1109/EMBC.2018.8513562. PMID: 30441699.

Jeong DU, Lim KM. Convolutional neural network for classification of eight types of arrhythmia using 2D time-frequency feature map from standard 12-lead electrocardiogram. Sci Rep. 2021 Oct 14;11(1):20396. doi: 10.1038/s41598-021-99975-6. PMID: 34650175; PMCID: PMC8516863.

Wickramasinghe NL, Athif M. Multi-label classification of reduced-lead ECGs using an interpretable deep convolutional neural network. Physiol Meas. 2022 Jun 28;43(6). doi: 10.1088/1361-6579/ac73d5. PMID: 35617943.

Barold, S. (2003). Willem Einthoven and the Birth of Clinical Electrocardiography a Hundred Years Ago. Cardiac electrophysiology review. 7. 99-104. 10.1023/A:1023667812925.

American Heart Association. (2023 April 5) “What Is an Arrhythmia?” Www.Heart.Org. https://www.heart.org/en/health-topics/arrhythmia/about-arrhythmia.

PhysioNet/Computing in Cardiology Challenge 2021: Program, access link: https://cinc.org/archives/2021/ on 07/12/2022

Stephen Butterworth. (1930). On the Theorie of Filter Amplifiers. Experimental Wireless and the Wireless Engineer, Bd. 7, S. 536–541.

Goutte, Cyril & Gaussier, Eric. (2005). A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. Lecture Notes in Computer Science. 3408. 345-359. 10.1007/978-3-540-31865-1_25.

He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Deep Residual Learning for Image Recognition. 770-778. 10.1109/CVPR.2016.90.

Elman, J. L. (1990). Finding structure in time. Cognitive Science, 14(2), 179–211. https://doi.org/10.1207/s15516709cog1402_1.

Hochreiter, Sepp & Schmidhuber, Jürgen. (1997). Long Short-term Memory. Neural computation. 9. 1735-80. 10.1162/neco.1997.9.8.1735.

Ian Goodfellow, Y. Bengio, A. Courville. (2016). Deep Learning. MIT Press. http://www.deeplearningbook.org.

Lecun, Yann & Bottou, Leon & Bengio, Y. & Haffner, Patrick. (1998). Gradient-Based Learning Applied to Document Recognition. Proceedings of the IEEE. 86. 2278 - 2324. 10.1109/5.726791.

Jonas Gehring, Michael Auli, David Grangier, et al. (2017). Convolutional Sequence to Sequence Learning. In: CoRR abs/1705.03122. arXiv: 1705.03122

Piero Esposito. (2020, May 24). Building a LSTM by hand on PyTorch. Towards Data Science. Access link: https://towardsdatascience.com/building-a-lstm-by-hand-on-pytorch-59c02a4ec091

Siu, David & Tse, Hanh. (2008). Hypertension and Cardiac Arrhythmias: a Review of the Epidemiology, Pathophysiology and Clinical Implications. Journal of human hypertension. 22. 380-8. 10.1038/jhh.2008.10.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. (2019). On the convergence of adam and beyond. arXiv preprint arXiv:1904.09237