FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

CNN-LSTM-based models to predict the heart rate using PPG signal from wearables during physical exercise

Lucas Ribeiro

DISSERTATION

Master in Informatics and Computing Engineering

Supervisor: Tânia Maria Pereira Lopes, Hélder Filipe Pinto de Oliveira.

October 31, 2022

CNN-LSTM-based models to predict the heart rate using PPG signal from wearables during physical exercise

Lucas Ribeiro

Master in Informatics and Computing Engineering

October 31, 2022

Abstract

Atrial fibrillation, or AFib is the most common form of arrhythmia. In fact, 3% of people over the age of 20 suffer from this condition and more shockingly, it is found that patients with arrhythmias are 5 times more likely to have a stroke [14]. These events of irregularity in the heart beat occur briefly and can be very sporadic which leads their detection to be rather cumbersome, with the standard diagnostic procedure being a long term continuous ECG. This leads to multiple problems, first of all, the ECG is commonly performed as the person is laying down in a hospital bed, which immediately distances the test environment from the real world scenario of living with AFib or another kind of arrhythmia, especially since arrhythmias are more likely to manifest during the practice of physical exercise.

From this need arises the alternative of using a PPG (Photoplethysmography) signal, which is an optical method of measuring the blood volume in surfaces such as the finger tip, wrist or ear lobe [14] and can be present in many portable devices like fitness bands and smartwatches, therefore enabling it to be used during the practice of physical exercise [13]. This alternative heart rate monitor is substantially less invasive and more mobile but it is also much more susceptible to motion artifacts.

However the motion artifacts that create this noise can be quantified through the pairing of an accelerometer to this device, which provides us with data regarding the acceleration of the devices over the 3 axis.

In this project, we used a model with CNN-LSTM architecture, which can be more easily imagined as two sub-models in a larger sequential model. The first of those sub models, the CNN, is used for feature extraction, while the second, the LSTM, makes it so that the features extracted become time dependent [12]. These kinds of models have been widely applied for video description, where the model interprets the sequences of events in a video, outputting a description of it, which in a way is very similar to the purpose of this project, where we expect the model to be capable of interpreting a sequence of data and outputting a single value, the heart rate, that, in a way, can be seen as a descriptor of that data.

The highest quality results achieved in this work were reached when utilizing a model such as the one that was just described, while pairing PPG and acceleration data in order to use them as its input and it managed to reach a mean absolute error of just 3.957 ± 4.272 bpm when running solely on the competition data provenient from the 2015 SCP dataset and 7.033 ± 5.376 bpm when running on the entire dataset, as well as 9.520 ± 8.443 bpm when running on a validation dataset.

Keywords: Deep Learning, Motion Artifacts, Artificial Intelligence, Photoplethysmography, Convolutional Neural Network, Long Short Term Memory

Acknowledgments

As this chapter of my life comes to an end, there's obviously a lot of people I crossed paths with and some of them were essential to the completion of this project.

Starting from those who guided me troughout this dissertation, I'd like to thank Tânia Pereira and Hélder Oliveira.

I'd also like to thank every one of my friends who had to endure every emotional meltdown I had over random acronyms and abbreviations like PPG, LSTM and CNN. Especially for pretending like it was interesting to them in the slightest bit.

I'd also like to thank both my parents for ensuring me it was all gonna work out, no matter how much I had going on in my life.

Finally, I'd like to thank my girlfriend. Without her, I would have most likely lost my mind trying to fit every one of my dreams and aspirations in under 24 hours. Forgive me for the sudden change to Portuguese, but as her mom so often tells her: "Quem tem uma Inês tem tudo, quem não tem uma Inês, não tem nada".

Lucas Ribeiro

"Give me everything I ever need
Or just enough so I can go to sleep
Well is it me or is it you who came to see
The scene when all those warm jets swallow me"

Nick Rattigan

Contents

List of Figures

List of Tables

Abbreviations

AAE	Average Absolute Error
CNN	Convolutional Neural Network
DL	Deep Learning
ECG	Electrocardiogram
HR	Heart Rate
LSTM	Long Short Term Memmory
ML	Machine Learning
MA	Motion Artifacts
MAE	Mean Absolute Error
PPG	Photoplethysmograph
RMSE	Root Mean Squared Error
SP	Spectral Peak
SSA	Singular Spectrum Analysis
SSR	Sparse Signal Reconstruction
SVM	Support Vector Machine
STD	Standard Variation

Chapter 1

Introduction

1.1 Motivation and Context

According to the World Health Organization, one third of deaths are caused by cardiovascular diseases, out of which 85% are attributed to heart attacks and strokes. When added to the fact that 3% of the people over the age of 20 suffer from atrial fibrillation [14], a specific kind of arrhythmia which, when present in a subject, is said to make them 5 times more likely to suffer a stroke, being the number 1 preventable cause of stroke [6], it is easy to see why the early detection of cardiovascular anomalies is essential to reduce the cardiovascular disease burden on global health. Over the last decade, wearables have risen to tremendous popularity among the general population, with 15% of tech customers in the United States already owning and regularly using wearable technology, such as smartwatches and fitness bands.
Following this, e-health welcomed a whole new array of opportunities for the continuous monitoring of the bio signals of each user. One of these devices is capable of monitoring a wide range of different medical risk factors, being equipped with dozens of sensors at times, from altimeters to thermometers and is capable of gathering all kinds of data, from motion to sound or even heart rate [15].

In order to diagnose this sort of condition, the close monitoring of the biosignals of a patient is necessary, with the most prevalent indicator being of course the heart rate, with the long and continuous ECG being the standard procedure, though the very nature of this events, which can be very sporadic, makes their detection very cumbersome.
As an alternative to this comes the PPG, a sensor commonly found in wearable devices that reached notoriety as a cheaper, far more mobile and less invasive alternative to the ECG, which regularly requires the subject to stay in bed, stationary whereas the PPG can be used under virtually any circumstance, per example, during physical exercise. Despite this, if the main advantage of the PPG is that it allows the subject to move, it is also this movement that leads to its biggest disadvantage, its predisposition towards motion sensitivity, this is a lower signal-to-noise ratio, as shown in figure 1.1.

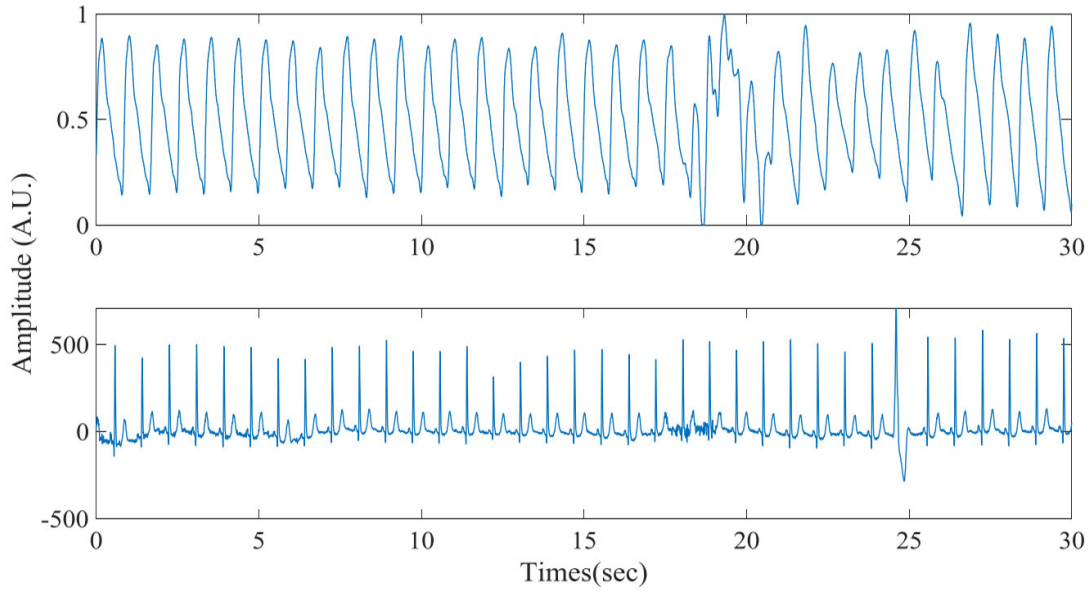

Figure 1.1: PPG signal contaminated by noise above (Time = 20 sec) vs an ECG signal in the same circumstances, extracted from [14].

From this issue arises the focus of this dissertation, which is the need for a strategy that can enable the accurate estimation of the heart rate, through a PPG sensor even when its signal is being heavily affected by motion artifacts and we have come to believe that a Deep Learning approach could be especially well suited as a solution for this problem.

1.2 Objectives

We have found that despite the widespread use of Deep Learning models for similar tasks, such as video description, the range of the approaches implemented with the goal of solving our problem is not as extensive as it could be. Besides this, there is also a lack of multi modal approaches, that combine the PPG data with other kinds, such as acceleration data, in hopes of providing the model with more information regarding the motion artifacts that affect the signal.
Therefore, the goal of this dissertation is to develop a Deep Learning-based model, capable of accurately estimating the heart rate, through a PPG signal and the corresponding accelerometer data, even when the subject is under extreme physical stress.

1.3 Contributions

The main goal of the work developed in the scope of this dissertation is that in some form it may contribute to the state of art. It is believed that trough the exploration of several key aspects of heart

rate estimation, it can provide a more in depth look at how to deal with the presence of motion artifacts in the signals available, as well as asserting how other relevant metrics may influence the results. In this process, the most relevant contribution of this thesis would be to conclude whether or not and to what degree the use of acceleration data could be useful to the aforementioned goal. This dissertation should provide more conclusions regarding the best way to structure the data and the kind of models that provide the best results and the influence of the difference window sizes used to obtain each label etc.

1.4 Structure Of The Document

This document is organized in the following way: The first chapter explains the motivations, context, objectives and contributions as well as, of course, the structure of the document. The second goes over the background of this research and contextualizing the most relevant concepts that preceded it while the third chapter explains many of the fundamental concepts applied within this paper.

Then the fourth chapter is the literature review, where many of the works related to this one are analyzed, extrapolating any information that could be useful within this context. This is followed by a chapter going over the methodology applied in this paper, describing the relevancy of the stages of this project and their order.

Finally, there is a chapter discussing the results that were obtained as well as one more laying out all the conclusions as well as possible steps to take following the term of the project.

Chapter 2

Background

This chapter has the purpose of supplying the reader with an overview of some of the concepts and events that led to this project. From the merging of health and technology, to the integration of certain pieces of hardware into our daily lives and a deeper look into how some of the relevant hardware related to this study came to be.

2.1 e-Health

What started in the early 2000's as a mere buzzword is now a part of the everyday lives of many people. With the world becoming more and more digital, the same sort of convenience that began being associated with other everyday activities started being demanded for the field of health as well, and so it had to adapt, merging itself with the digital world for the creation of the concept of e-health.

> *"e-health is an emerging field in the intersection of medical informatics, public health and business, referring to health services and information delivered or enhanced through the Internet and related technologies. In a broader sense, the term characterizes not only a technical development, but also a state-of-mind, a way of thinking, an attitude, and a commitment for networked, global thinking, to improve health care locally, regionally, and worldwide by using information and communication technology."*
>
> - Gunther Eysenbach [4]

This new field began being described by its efficiency, accessibility, its cost-value proposition, its evidence-based approach to diagnostics and its expansiveness, as it more and more led to the expansion of the field of healthcare beyond its traditional boundaries, opening the way for, per example, the gamification and automation of diagnostics.

As a final note, it should be said that it also expanded upon the ethical concerns already associated with health care and the disparity in the access to it, as the surge of e-health also served as a way to bring the digital divide into the field.

2.2 Wearables in Medicine

Once known as wearable devices, they have now become such an integral part of our daily lives, that the single word "wearables" is enough for anyone to be fully aware of the subject in question. But what are wearables? They are any electronic gadget or device that can be worn, used as an accessory, glued to or even implanted in the body. These commonly carry a lot of sophisticated sensors which might be permanently, wirelessly, connected to the web.
Its use cases can be seen as seemingly infinite. If at first early adopters were among the military and space programs, nowadays every "average joe" could own one and so they began being adapted to more common use case scenarios. With so much data being gathered by them, as shown in figure 2.1, assessing several of the biosignals of the user hundreds of times per second, the adoption into the field of health was predictable.
From just informing the user that their medication is due to much more complex tasks like automated diagnostics, their ever-evolving technology not only shows promise that it can lead people towards a more healthy lifestyle but it also provides the possibility for one more step towards the digitization of healthcare [20].

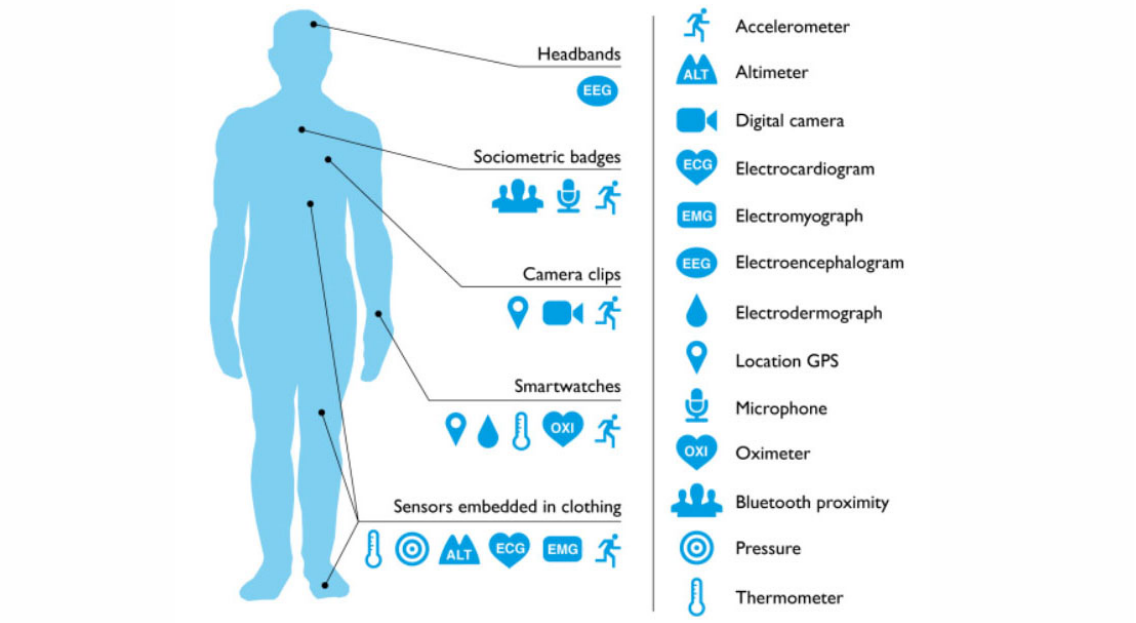

Figure 2.1: Representation of all the sensors that can be made available through the use of wearable devices, extracted from [15].

Wearables offer an undeniably impressive value proposition in this field, as with the addition of just one small device to your daily life you can have the possibility for continuous tracking of

several medical risk factors, the possibility of the early detection of a disease and the reduction of healthcare expenses [20].

2.3 The PPG Signal

The PPG, or plesthysmography, signal has risen in popularity as an alternative to the traditional ECG. While ECG measures the heart rate based on the bio-potential generated by electrical signals that control the expansion and contraction of heart chambers [18], the PPG is an optical method that uses LEDs and a photo detector to measure the change in the volume of blood going through its vessels, and does not require the attachment of any sensors to the subjects skin and allowing for it to be used in different body parts like the fingertips, wrists and earlobes [9] . The arrangement of the LED and photodetector can be one of two, depending on the kind of PPG used, either transmittance or reflectance, as shown in figure 2.2, in which the two parts are placed on opposite sides of the tissue or on the same side, respectively.

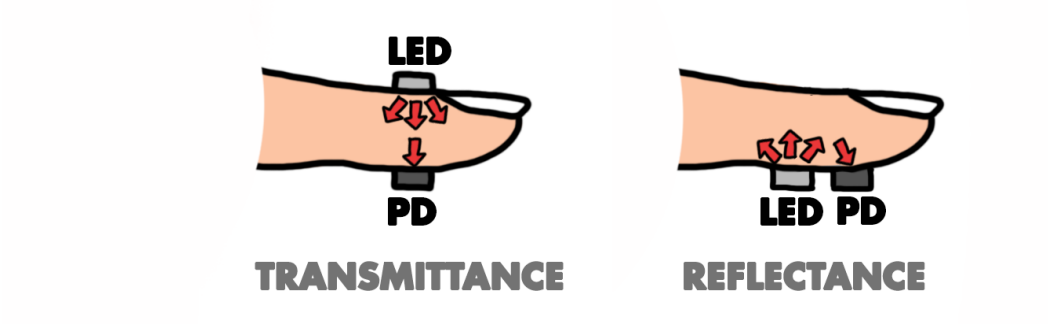

Figure 2.2: Representation of the two different kinds of PPG sensors, extracted from [7].

Once the LED emits light, it penetrates the tissue and is primarily absorbed by it, with the remaining light being reflected and/or transmitted and then detected by the PD, which then computes it into a quasiperiodic signal [5] resemblant of the fluctuation in volume of the pulsatile component of the blood, which are consequences of the systolic and diastolic phases of the heart beat and then able to be used for the estimation of the heart rate, as shown in figure 2.3.

However, during the execution of intense physical activity, the different motions performed by the subjects can lead to changes in the pulsatile component of the blood, making the representative signal less periodic and less correlated with the actual heart rate, which clearly becomes a serious issue if the sensor is being used with its estimation as the end goal.

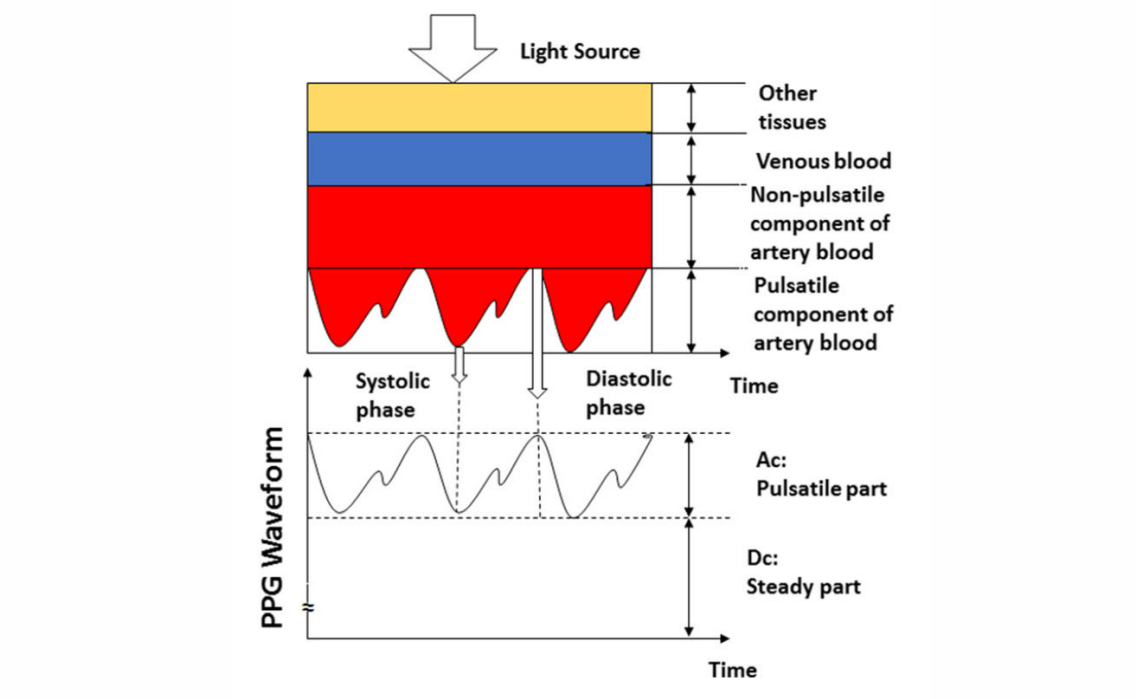

Figure 2.3: Representation of the transcription of the pulsatile component of the blood into a PPG signal, extracted from [7].

2.4 Summary

With the seemingly intertwined rise of e-Health and wearable, a new window for improvement upon already in use medical practices is open and so the PPG arrives as a more mobile alternative to the traditional ECG, however due to this same mobility it is exposed to motion artifacts which deteriorate the quality of the signal and make it seemingly useless when the subject is performing intensive physical exercise. In order to solve this issue, there is a need to quantify this motion and de-noise the signal, improving the accuracy of the heart rate estimation obtained trough the sensor, which is where the subject of this thesis comes in.

Chapter 3

Fundamentals

3.1 Deep Learning

Right from 2006, when Geoffery Hinton coined the term "Deep Learning", a new subset of Machine Learning began taking its first steps. The term refereed merely to a new kind of neural network, which makes it especially surprising that the term took so long to be created, considering the first neural network had been created nearly 50 years earlier by Frank Rosenblatt, famously referred to as the Perceptron, as represented in figure 3.1. [8]

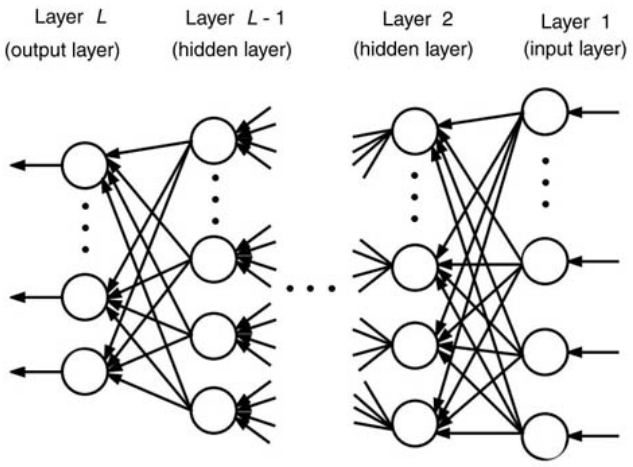

Figure 3.1: Representation of a multi-layer perceptron, extracted from [16].

Nowadays, Deep Learning is defined as subset of machine learning algorithms capable of greatly outperforming traditional machine learning approaches in their ability to interpret raw data [19]. For a long time, building a Machine Learning model demanded in-depth expertise in whichever scientific field the data was obtained from in order to build a suitable feature extractor

that could accurately transform the raw data into a representation that the model was able to interpret.

With Deep Learning, this problem is mostly a thing of the past, feature extraction as become yet another automatic stage in the process. Deep Learning is often characterized as a layered approach since it relies on levels of representation to produce results. The first level of representation is obtained from the raw data and from there on out, each level is obtained from the previous one in higher and higher degrees of abstraction. These "layers" can be combined in endless ways and in the right disposition, increasingly complex tasks can be learned, as figure 3.2 suggests.

Let's focus on classification problems for a moment. For each input, every layer attempts to amplify the features that make the subject more identifiable and suppress any that are deemed irrelevant. If you consider the common problem of identifying characters based on pictures of someones handwriting, the first layer will most likely identify the lines that make up a character, feeding that data to the following layer, which will try to identify patterns in how those lines are combined in each picture. This sort of process is repeated over the remaining layers until, in the end, the feature arrangement of each character would be known, or, at least, accurately determined most times, allowing for the solution of the problem at hand.

These characteristics make Deep Learning especially useful for high-dimensional data, which could be hard to visualize for humans. This sort of data can be found in a large number of fields, but is most often useful when applied to image and speech recognition.

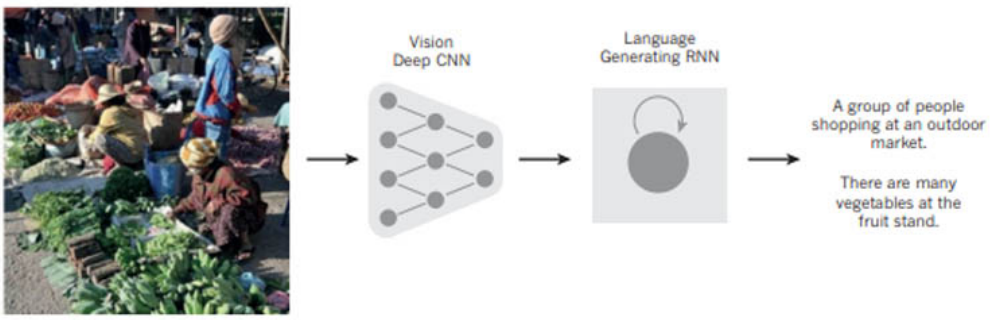

Figure 3.2: An example of image captioning trough a CNN and LSTM based Deep Learning model similar to those employed in this project, extracted from [19].

3.2 Neural Networks

Often also referred to as artificial neural networks, these structures attempt to mimic how the electric impulses of neurons propagate trough the brain, strengthening certain pathways and disregarding others in a process that in every day life we call learning. Just like the neurons in a brain, divided into dendrites, nucleus and axon, ANN's are also divided into 3 main parts, the input layer, the hidden layer and the output layer, as represented in figure 3.3 [8].

Regardless of all of that, objectively, Neural Networks are mathematical structures. Their structure can be divided into its processing elements, the neurons, and the connections between them. Each connection having a weight associated to it and allowing each neuron to receive stimuli from the ones connect to it. This is where the distinction between the three layers comes from, with the input layer being comprised of all the neurons that receive stimuli from outside of the NN, while the neurons whose output is not connected to another are referred to as output neurons, with the one that are only connected to other neurons comprising the hidden layers.

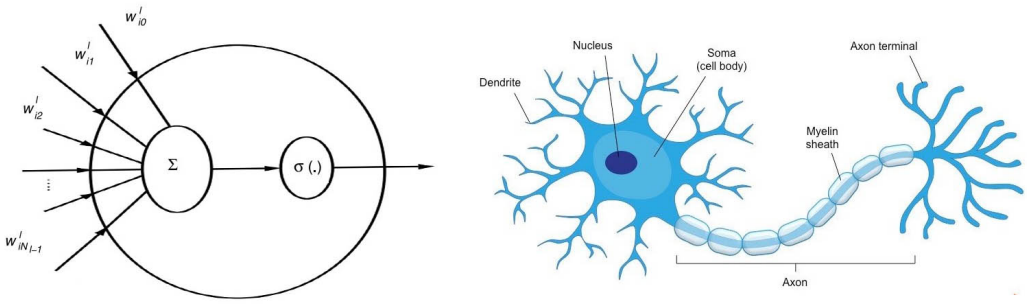

Figure 3.3: Comparison between an artificial hidden neuron and a representation of a biological neuron, partially extracted from [16].

The process of learning happens trough the search of an optimal state of the weight parameters, referred to as training. At each iteration, the error is determined, the error being the difference between the actual outputs of the neural network and the desired outputs and then the weight parameters are adjusted with the goal of minimizing the error.
However, there are different ways of connecting these neurons to one another and different ways to interpret stimuli, which allows for a diverse range of different types of neural networks, with Convolutional Neural Networks and Long Short Term Memory Neural Networks, being the two main kinds associated with this project [16].

3.3 Convolutional Neural Networks

Convolutional Neural Networks are defined as a kind of ANN made up of, of course, convolution layers, but also max pooling layers and fully connected layers, with the purpose of identifying features automatically and hierarchically.
These kinds of networks have achieved incredible results in the field of computer vision, often managing to attain the same level of accuracy in classification problems as experts of the corresponding scientific fields.
CNN's are especially suited for data that can be displaying in the form of a grid, as represented in figure 3.4. Taking images as an example, we can say that the convolution layer of the CNN contains what is called a kernel, a grid of parameters that operate upon the pixel grid of the image, optimizing themselves trough back propagation in order to identify the image's dominant features

and since these layers can be stacked, the identified features can also be more and more complex [3].

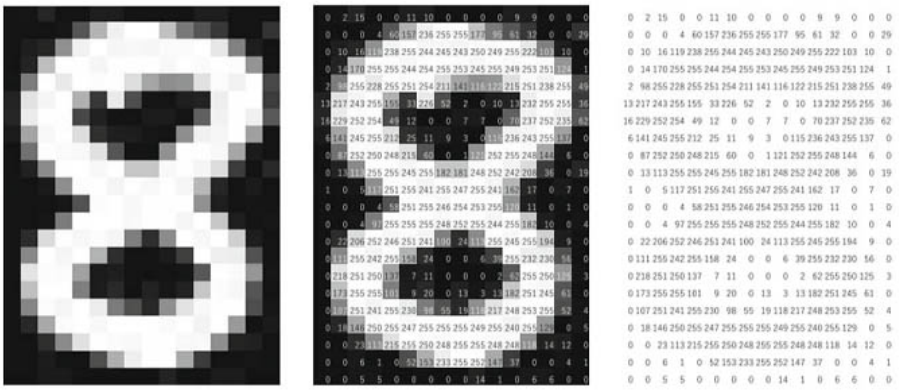

Figure 3.4: Representation of how an image can be displayed as a grid, extracted from [3].

3.3.1 Convolution Layer

In order to understand the inner working of a convolution layer, it is essential to understand what convolution, as a mathematical operation, is. Convolution is a linear operation, where a small grid of numbers, the kernel, operates over the input, with an element-wise product (dependent on the selected activation function) being calculated at every possible overlap between the two, obtaining an output value that is saved on what is known as a feature map, as represented in figure 3.5.

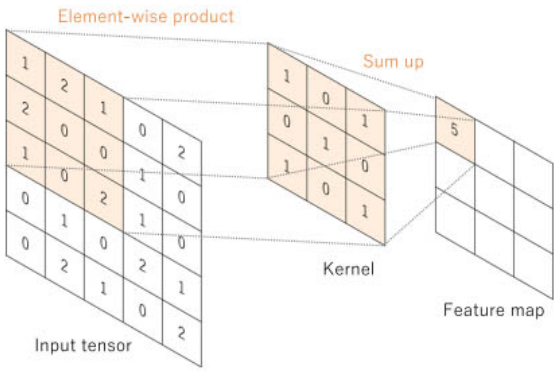

Figure 3.5: Representation of a how a single value of the feature map is obtained from the input, extracted from [3].

3.3.2 Pooling Layer

In a CNN, the pooling layer, more than often a max pooling layer, is adjacent to the convolution layer. Its purpose is to reduce the size of the feature map and to decrease the relevance of small distortions that can be detected by the convolution, as represented in figure 3.6.
Max pooling specifically works by selecting subsections of the feature map and selecting the max value in them as the output.

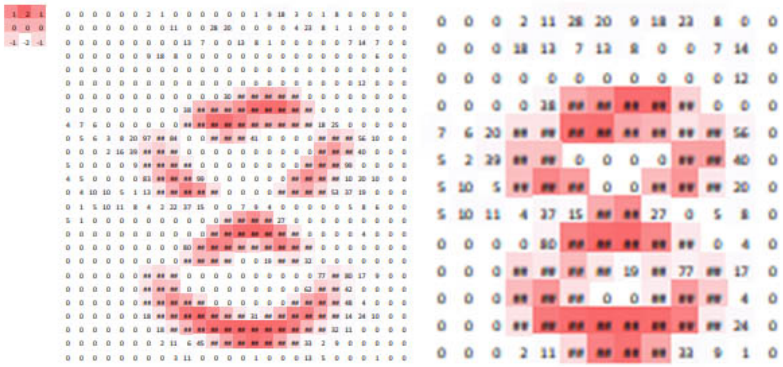

Figure 3.6: Representation of a how a max pooling layer with a 3x3 filter, operates over its input, reducing its dimensionality, extracted from [3].

3.3.3 Fully Connected Layer

The final layer in a CNN is usually the Fully Connected Layer. This layer starts by flattening the feature maps output by the previous layer and connecting every one of them to a respective dense layer, typically, providing an output node for each possible output. Per example, in the common handwritten character identification problem, the number of output nodes in the CNN would be 10, one for each digit from 0 to 9.

3.4 Long Short Term Memory

LSTM's are a sub-genre of Recurrent Neural Networks, which distinguish themselves over the fact that some of their neurons are connected to themselves in a sort of loop. This is extremely helpful when dealing with sequential data, in which the ability to retain information regarding consecutive data points enabled by those extra connections can be key.
However, even though RNN's outperform ANN's when it comes to dealing with sequences, they still struggled when long term dependencies were at stake due to what is reffered to as the vanishing error problem in [10]. This problem arises from the weight update being especially slow

in this kind of network and it led to the need for the creation of an alternative suited to problems where this sort of issue could be a deal breaker. The alternative would be the LSTM.

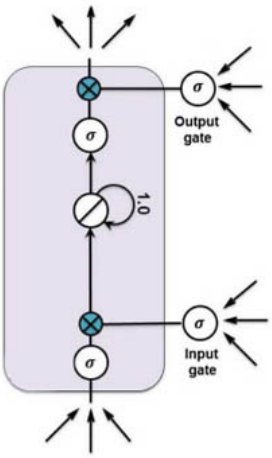

Figure 3.7: Representation of an original LSTM unit, extracted from [11].

The LSTM differentiates itself by having multiple gates, responsible for controlling the data flow in each cell. Originally LSTM cells had two gates, the input and the output gate, as seen in figure 3.7. The input gate was responsible for acting upon the input of the cell, while considering the importance of the output from the previous time step, while the output gate was responsible for stopping irrelevant data from affecting the following cells. In the middle of all of this was, of course, the memory cell, which retained the states of previous cells, depending on the actions of these gates. However the configuration of LSTM cells has evolved over the years, with the highlight being the addition of an extra gate, known as the forget gate, which is capable of more drastic measures, erasing the content of the memory cell once activated.

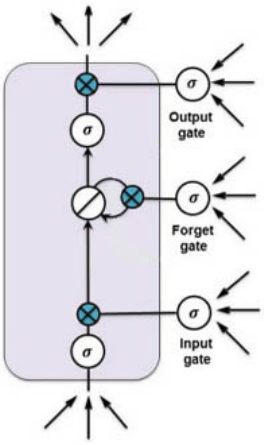

Figure 3.8: Representation of an LSTM unit with a forget gate, extracted from [11].

Chapter 4

Literature Review

Within this review, the analyzed pieces of work will be divided into 3 different categories, depending on the style of the approach used, namely "Traditional Statistical-Based Motion Artifact Subtraction and Spectral Peak Estimation", "Machine Learning Signal Segment Classification and Subsequent Heart Rate Estimation" and "Deep Learning Estimation of Heart Rate And The Possibility Of Transfer Learning", as well as one extra paper evidentiating the advantages of Deep Learning over Machine Learning.

First there will be an overview of the methodology used in each paper, then there will be a section for discussion that will focus on the advantages and disadvantages of each approach, as well as comparisons between them. Finally, a summary of the analysis will be at the end.

4.1 Methodology

To create this literature review, the Scopus citation database was used to search for papers that could be relevant to the topic of this dissertation. Then upon researching the contents of each paper that had been deemed relevant, six were selected to be discussed in this review, first being summarized in a table that highlights their authors, pre-processing techniques, the datasets used, their methodology and finally the quality of their results. Before getting into the body of this document, a small overview was written in order to offer a landscape look at the studied work.

The review was separated into six points of discussion, first the datasets used (most common, similarities, outliers etc.), then different approaches to the pre-processing of the data was discussed, as well as the algorithms used and how they differentiated themselves in terms of their effectiveness. Finally, both common shortcomings identified among all the approaches were discussed, followed by a quick summary of the findings accomplished across the selected papers.

4.2 Traditional Statistical-Based Motion Artifact Subtraction and Spectral Peak Estimation

For the scope of this review, only one study with a statistical-based approach was selected. Zhang, Z. et al. (2014) [23] is an heavily referenced and very influential paper in the field of heart rate estimation, with around 400 citations as of today, according to Scopus.

In this study, a framework known as TROIKA, as shown in figure 4.1, was created for accurate heart rate monitoring through a wrist-mounted PPG signal and accelerometer, during intensive physical exercise. This framework was divided into three parts: denoising, spectrum estimation and finally peak selection and verification.

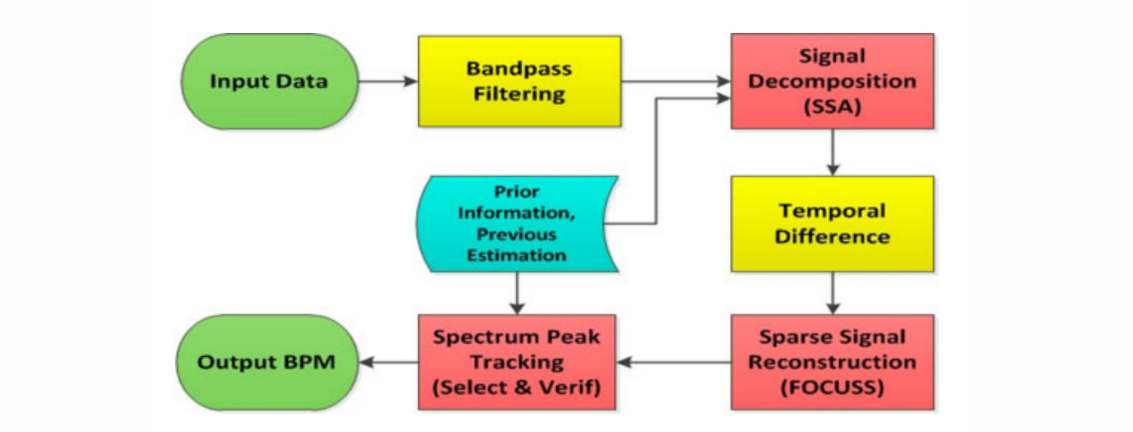

Figure 4.1: Flowchart of the TROIKA framework, extracted from [23].

After passing the signal through a standard band-pass filter, with the intent of removing outlier artifacts outside of the relevant frequency range, the time series was decomposed into oscillatory components and noise, through the use of a Singular Spectrum Analysis.
In order to conclude the denoising process, the framework uses the available acceleration data in order to determine which of its components is a result of the motion of the subject. In order to do this, the Periodograms of the acceleration signal must be calculated in order to obtain its dominant frequencies and remove them from the decomposed PPG signal.
Then the signal is temporarily differentiated, as a way to evidentiate its periodic components, suppressing the surviving motion artifacts before employing a Sparse Signal Reconstruction Algorithm. Finally they proceed to the final step, Spectrum Peak Tracking. In it, this study exploited the fact that HR estimations for two consecutive time windows tend to be approximate, using an initialization stage, in which the subject is told to keep their hands still, so the initial HR can be estimated by picking the highest spectral peak.

For each new search range, 3 peaks higher than a given threshold were selected, from which the final estimation was obtained following 3 simple rules:

- if one of them has an harmonic relation to the previous estimation, it is picked.

- if there is no such relation, the one with the smallest variation when compared to the previous estimation is picked.

- if there is no peak above the threshold, the previous estimation is maintained.

Finally for verification, regularization was employed if the variation between two consecutive estimations was too large and in the case that the threshold is not met for a given number of consecutive windows, for which the trend was taken into account to improve the results.

4.3 Machine Learning Signal Segment Classification and Subsequent Heart Rate Estimation

For this chapter two different papers were taken into consideration, Zhang Q. et. al. (2017) [22] and Bashar, S. et al. (2019) [1]. The first of which focuses on the creation of a Machine Learning based framework for the estimation of the heart rate, as shown in figure 4.2, while the second focuses mainly on feature engineering.

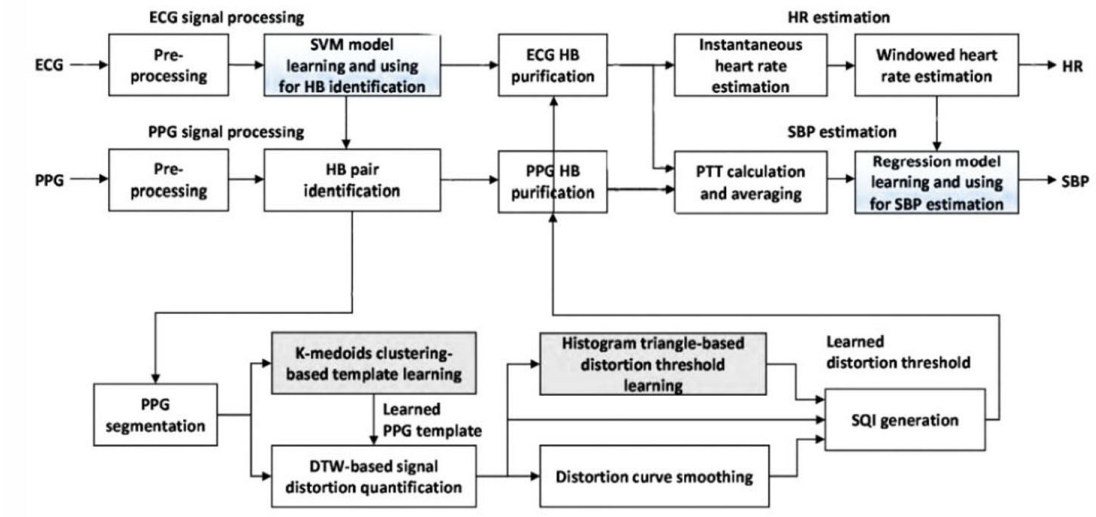

Figure 4.2: Flowchart of the framework present in Zhang. Q. et al (2017), extracted from [22].

In Zhang Q. et. al. (2017) [22] the framework developed is mainly divided into 3 stages: the first starts by dividing the ear-ECG signal into segments that are considered heartbeat candidates and then utilizing the data from the chest-ECG as the ground truth for training a Support Vector Machine to identify the raw heartbeats. The second stage puts those same segments through a k-medoids clustering algorithm in order to create a template of a high quality signal, which is then used as the term of comparison to measure the distortion observed in the raw PPG signal segments. These measurements are then used to determine a number of thresholds, that are used to determine Signal Quality Indexes and progressively increased until a given number of segments

can be labeled as those of high quality. Finally, the indexes are used to estimate the heart rate in any given window.

On the other hand, in Bashar, S. et al. (2019) [1], though not as much information is available about the steps taken towards the estimation of heart rate, given their focus on feature engineering, it is known that their focus is to achieve accurate estimation without a motion artifact removal stage.

To determine the most suitable feature set for the study, they ran 7 different sets of features to their framework, in order to compare the quality of the results between them.

The framework is divided into 2 parts: First a k-means clustering algorithm divides the signal into noise and non-noise data, as shown in figure 4.3.

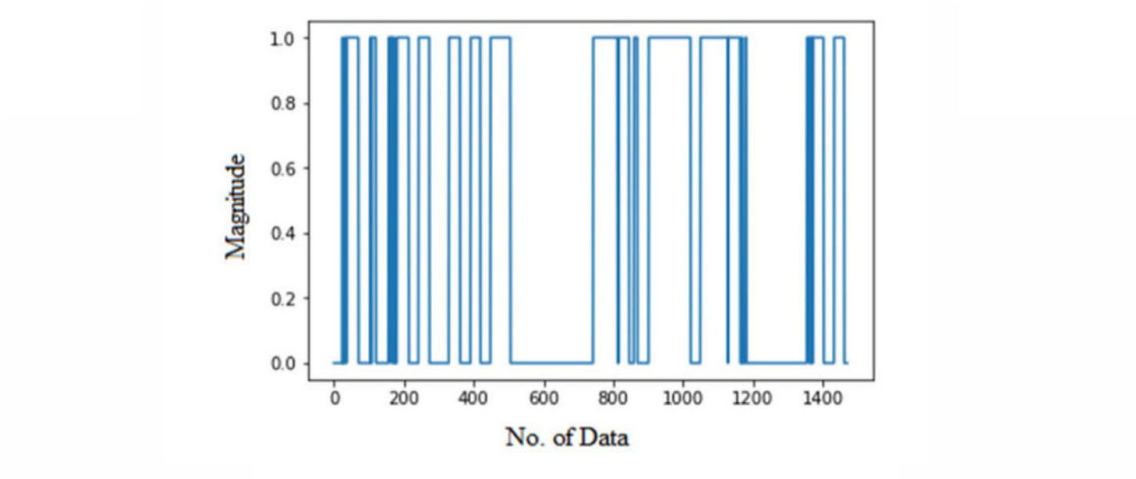

Figure 4.3: Representation of the division of the data points in noisy (Magnitude=1) and non-noisy (Magnitude=0) data after k-means clustering, extracted from [1].

Once this is done, a Random Forest Regressor algorithm trained on each set of data separately, in order to obtain a heart rate estimation model.

In the end it was determined that among the 7 different sets of features, the most suitable for the non-noisy data was the one corresponding to the peak positions of the PPG and the acceleration signal after spectral subtraction, while for the noisy data, the most suitable set was the one that used the peak power of the PPG signal and the peak position of the acceleration data.

4.4 Deep Learning Estimation of Heart Rate And The Possibility Of Transfer Learning

For this chapter, Biswas et al. (2019) [2] was the first paper to be analyzed. The methodology proposed in this study, as shown in figure 4.4, utilized first a Convolutional Neural Networks for automatic feature extraction, then Long Short Term Memory layers to enforce time dependency in the data, in order to make up for the fact that the features are not completely phase invariant and

depend on the offset between the beginning of the sample and the first heartbeat and, finally, on a layer with a single neuron with a linear activation function.

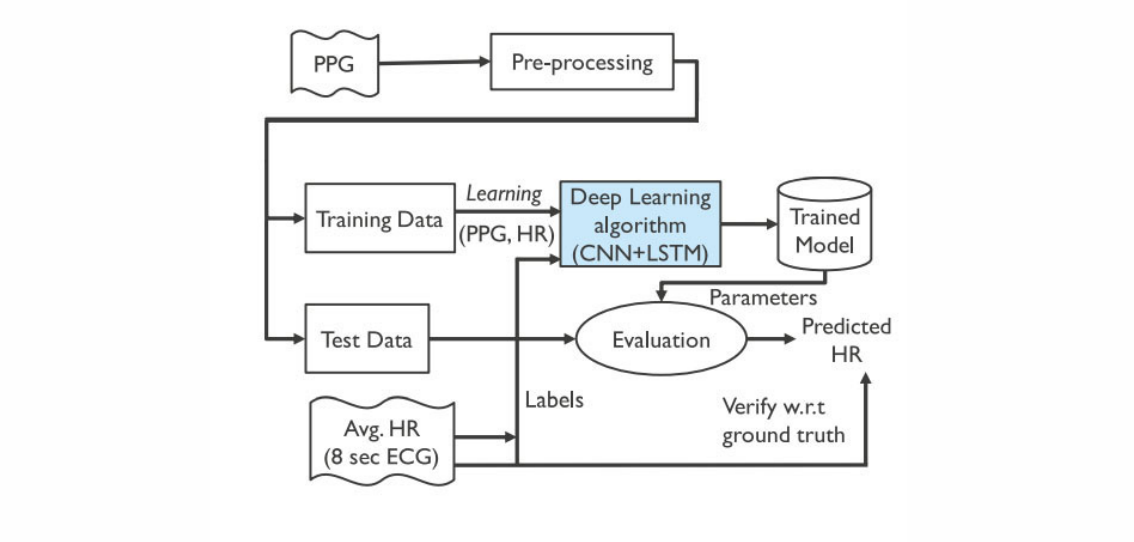

Figure 4.4: Representation of the methodology utilized in Biswas et al. (2019), extracted from [2].

The other paper in question was Shyam et al. (2019) [17]. This paper built upon the work done in Biswas et al. (2019) [2], finding that the requirement of the paper for patient specific retraining, as shown in figure 4.5, was too cumbersome during the actual deployment of the model and should be improved upon.

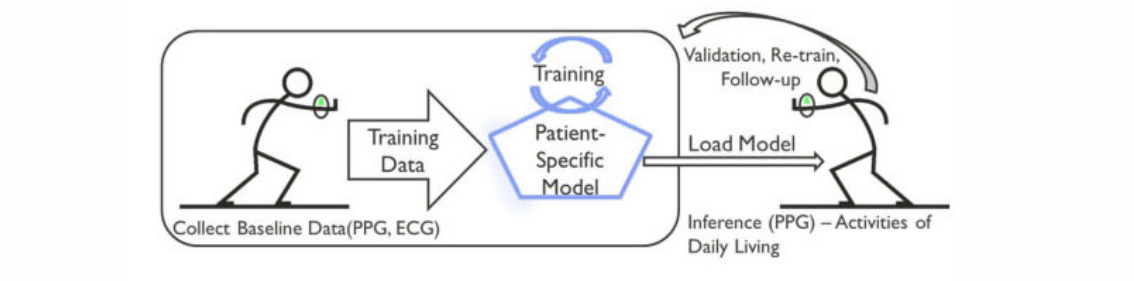

Figure 4.5: Use case scenario for the framework developed in Biswas et al. (2019), with focus on the need for patient-specific retraining, extracted from [2].

In order to achieve this, they set on to assess the feasibility of transfer learning in this scenario. First they created a similar framework, also using Convolutional Neural Networks and Long Short Term Memory Layers, with the main differences coming first in the way they divided the data at the start, dividing each eight second time window into eight one-second time-window, which were then provided to an inception block with five convolution layers with different kernel sizes. Alongside this, the one second time-windows were also fed to an LSTM layer, in order to obtain time-dependent features. Then, all the feature representations obtained from these two blocks were concatenated and fed to another LSTM layer and a final fully connected layer that outputs the predicted heart rate.

Then, they evaluated this model, taking advantage of their proprietary ADI dataset, in four different scenarios:

- A model, exclusively trained and evaluated on the ADI dataset.
- Another trained on the SPC dataset but evaluated on the ADI dataset.
- Validation on the ADI dataset, with pretrained model weights on the SPC dataset and optimization of the Long Short Term Memory and Linear blocks.
- Similar to the previous one but retrieving 15% of the training data from the ADI dataset.

Among these conditions, the highest performing was the first. Out of the remaining, the second presented the worst results, while the third and fourth conditions obtained relatively good results, with the third reaching convergence substantially faster.
We should also note that despite the superior results, the first condition took much longer to converge than the alternatives. Due to this, the selected condition as the most favorable was the third, which obtained a Mean Absolute Error of 4.1 beats per minute. For comparison, when validated against the SPC dataset, it obtained a Mean Absolute Error of 3.36 beats per minute, while Biswas, D. et al. (2019) [2] achieved a significantly better 1.47 beats per minute.

4.5 Discussion

From the analysis to all the aforementioned papers, there are many conclusions that can be taken. First of all, it can be noted that despite the evolution of different approaches, it is still frequent for machine learning approaches to have inferior results to that of the most accurate traditional methods.
The same also happens if the focus is put solely in Machine Learning, with both Deep Learning approaches getting inferior results to Zhang, Q. et al. (2019) [22], which happens to be the approach with the highest quality results within the scope of this review, with its Average Absolute Error falling at just 0.8 beats per minute.

When it comes to the shortcomings of each set of approaches, there are many things that can be pointed out. Right from the traditional methods, in Zhang, Z. et al (2014) [23], there is a chance that upon removing the dominant frequencies associated with motion artifacts after the signal decomposition stage, that the frequency associated with the heartbeat may be removed as well, since the periodicity of certain common motions can match the heart rate. This is much less of a problem in other types of techniques that approach the problem without the need for motion artifact removal. Still, even under the scope of traditional methods there are relative solutions for this problem, such as removing the harmonic and fundamental frequencies associated with the heart rate from the set of frequencies associated with acceleration data.
Another problem related to this is that Sparse Signal Reconstruction, as the name suggests, can have very poor performance if the signal happens to be non-sparse, which is the case when there

are extreme and continuous hand movements. However, previous steps tend to sparsify the signal before it arrives at this stage.
When it comes to the Machine Learning based approaches, Zhang Q. et al. (2017) [22], despite providing the best results in this review, there is one notable shortcoming as the smoothing operation that the signal is put through can create difficulties in detecting distortions in it, which is why the raw data is still useful.
Finally, regarding the Deep Learnings approaches, there is of course the issue of the generated features being time invariant, but both papers in this review resorted to the enforcement of a time dependency through the use of a Long Short Term Memory layer, which of course can greatly reduce the effects of this phenomenon. Another main problem is that the deeper a Convolutional Neural Network gets, the more the early computations are disregarded as their significance exponentially decreases with the progression of the data flow. Still, these papers show great promise for Deep Learning approaches and both have their advantages, as per example, even though [2] gets better results, the way [17] used five parallel convolution layers with different kernel sizes can be more robust as it accommodates for a broader range of different feature representations.
In this paper, the objective is to analyze the performance of different deep learning approaches, as well as in comparison with Machine Learning techniques, for the quality assessment of the signal rather than the estimation of the heart rate.
The different approaches tested are as follows:

Machine Learning	Time-series based DL	Image-based DL
An SVM classifier with 42 features, from the temporal domain, spectrum domain, as well as statistical summaries of segments and non linear dynamic analysis, that replicated previous work.	ALSTM-FCL: characterized by a fully convolutional block, a shuffle layer and an attention LSTM layer.	VG19: a Convolutional Neural Network with 19 layers and 3x3 convolutional filters.
	A Fully Connected Network.	Resnet 18 and Resnet 50
		Xception

Table 4.1: Short description of all the models taken into consideration.

As shown in figure 4.6, SVM did not benefit from using the full extent of the training set. On the other hand, ResNet18 approached the accuracy level of SVM while using just 50% of the training set and completely outperformed any other algorithm once the full dataset was in play. Xception and VG19 also outperformed SVM with the use of the full dataset.

Another notable conclusion that stems from the work done in this paper is that it proves the feasibility of image-based approaches for this field. As, even though the problem at the end is not exactly the same, they are relatively similar and it can be observed that for the one at hand, image-based approaches like ResNet18 greatly outperformed the standard time-based approaches presented.
On another relevant point, the concept of transfer learning assessed in Shyam, A. et. al. (2019)

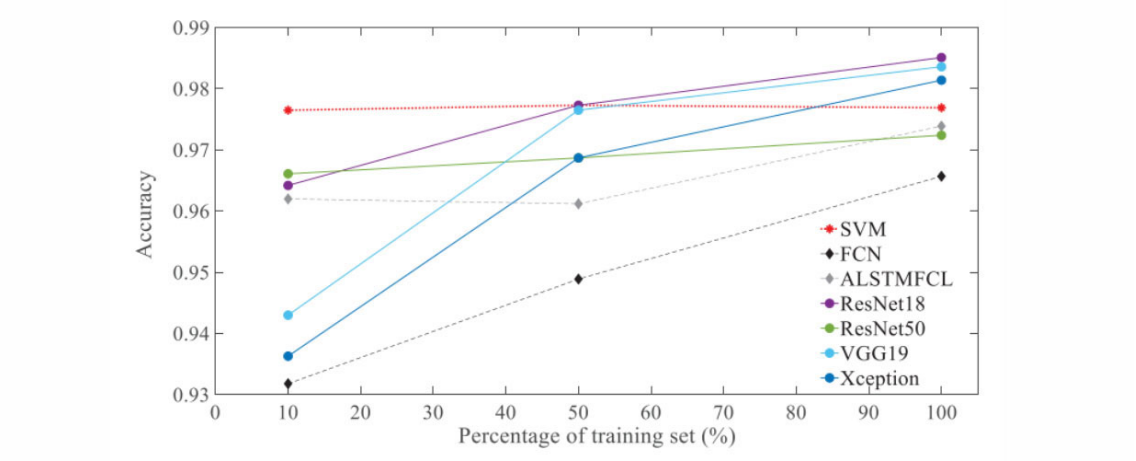

Figure 4.6: Learning curves for all 7 different approaches tested in Pereira, T. et al. (2019), extracted from [13].

[17] proved indeed feasible, obtained results with a Mean Absolute Error of 4.1 beats per minute, under its most favorable condition. For comparison, when validated against the SPC dataset, it obtained a Mean Absolute Error of 3.36 beats per minute, while Biswas, D. et al. (2019) [2] achieved a significantly better 1.47 beats per minute.

Out of the remaining conditions, the second presented the worst results, due to the existing variance between PPG devices used, while the third and fourth conditions obtained relatively good results, with the third reaching convergence substantially faster.

We should also note that despite the superior results, the first condition took much longer to converge than the alternatives.

Authors	Pre-processing	Datasets used	Methods	Results
Zhang, Z. et al. (2014) [23]	PPG and acceleration signals bandpass filtered from 0.4 and 5 Hz.	IEEE Signal Processing Cup 2015 + extra training and test set.	Singular Spectrum Analysis, Sparse Signal Reconstruction, Spectral Peak Tracking.	AAE: 2.34 bpm.
Bashar, S. et al. (2019) [1]	PPG signal filtered by 2nd order bandpass filter from 0.4 to 5 Hz.	IEEE Signal Processing Cup 2015 + extra test set.	K-means clustering, Random Forest Regression.	AAE: 2.81 bpm.

Authors	Pre-processing	Datasets used	Methods	Results
Zhang, Q. et al. (2017) [22]	Ear-ECG and PPG (resampled to 500Hz) signals filtered by Butterworth bandpass filter from 2 to 30 Hz and 0.5 to 8 Hz, respectively.	Unnamed proprietary dataset.	Support Vector Machine, K-medoids Clustering, Dynamic Time Warping, Histogram Triangle.	AAE: 0.80 bpm.
Shyam, A. et. al. (2019) [17]	PPG signal upsampled to 125Hz. PPG passed through a second-order band-pass Butterworth filter with cut-off frequency of 0.5-5 Hz. PPG normalized on a subject basis.	IEEE Signal Processing Cup 2015 + ADI dataset.	Convolutional Neural Network, Long Short Term Memory, Fully Connected Network and Transfer Learning.	AAE: 3.36 bpm AAE from Transfer Learning: 4.10 bpm
Biswas, D. et. al. (2019) [2]	PPG signal band-pass 4th order butterworth filtered from 0.1 to 18 Hz. PPG signal normalized from -1 to 1.	IEEE Signal Processing Cup 2015 + IMEC database.	Convolutional Neural Network, Long Short Term Memory and Final Dense Layer.	AAE: 1.47 bpm.

Table 4.2: Short description of each study within the scope of this review.

4.6 Datasets

Across the papers mentioned in the literature review, there were five different datasets mentioned. However, upon close inspection it was noted that all of them except for the IEEE Signal Processing Cup 2015 Dataset were either private, or had owners that were unresponsive upon contact.
Still, the IEEE Signal Processing Cup 2015 Dataset, just like Zhang, Z. et al. (2014) [23], is pretty much impossible to not come across when researching datasets for a project with the goal

of estimating the heart rate through a ppg-signal, as it is a benchmark in the field, being heavily utilized and referenced.
It is a dataset comprising two separate groups of subjects, the first one totalling 12 subjects and being commonly used as a training set, while the second totals just 8 and is often used as the test set. In this dataset the subjects were made to perform only arm movements and the data was recorded simultaneously and gathered from a wrist oximeter (two-channel PPG), accelerometer and single led ECG at a sample rate of 125Hz.
Outside of the spectrum of the literature review, we still took on the task of searching for other available datasets that could be helpful for our project, and we come across the following:
The BIDMC PPG and Respiration dataset, which recorded the PPG and ECG signals of 53 severely ill subjects, at a sample rate of 125 Hz and is available publicly and without any access restrictions at PhysioNet.
The IEEE TBME dataset, which contains the raw PPG signals of 42 different subjects, at a sample rate of 100Hz and labeled by an expert rater and a reference ECG signal. Though it is identified as a respiration dataset, this data can be useful for this project. This dataset is available publicly and without any access restrictions at Dataverse.
The MIMIC-III dataset was suggested due to its wide range of different clinical data and its public availability. To access it, there is only the need to be registered at PhysioNet and to have completed the CITI 'Data or Specimens Only Research' training program. Finally, the PPG-DaLiA dataset was suggested because of its similarity to the 2015 SCP dataset. Though it actually offers a much broader range of different biometrics, all the ones contained in the 2015 SCP dataset are also available and so it could be a great candidate to be paired alongside it. This data was extracted from 15 subjects, carrying out eight different kinds of day-to-day activities, such as sitting, using stairs, having lunch or driving a car.

Name	Nr Of Subjects	Data	Availability
IEEE Signal Processing Cup 2015 Dataset.	Training set: 12. Test set: 8.	Two-channel PPG. Tri-axial accelerometer. Single lead ECG.	Public.
BIDMC PPG and Respiration dataset	53	ECG and PPG.	Publicly available at PhysioNet.
IEEE TBME dataset	42	PPG, ECG and respiratory data.	Publicly available at Dataverse.
MIMIC-III	Not determined.	Wide range of medical data.	Available upon completion of training program.
PPG-DaLiA	15	ECG, accelerometer, inductive respiration sensor, temperature, electrodermal activity.	Publicly available at UC Irvine Machine Learning Repository.

Table 4.3: Short description of all the data sets taken into consideration.

4.7 Summary

As a final conclusion, though traditional methods still display competitive results, it can be said that within the scope of this review, the Machine Learning approaches seem to be the ones capable of deploying the more accurate results as of this very moment.
Despite this, Deep Learning shows the most potential for improvement, may it be through the growing access to larger datasets, the use of image-based approaches or through the possibility for transfer learning, as well as due to the availability for automatic feature extraction, which partially removes the need for highly advanced knowledge in the field of health care.
When it comes to the datasets, it seems the IEEE Signal Processing Cup 2015 dataset is by far the most interesting candidate, but there is also the possibility of pairing it with other datasets.

Chapter 5

Methodology

In this chapter, the main goal is to give the reader a complete and thorough overview at every single aspect that was part of or relevant to the work developed within this dissertation, going from the data and its characteristics, to the model that were created, how the results were obtained, its side experiments and even the structure of its workflow.

5.1 Database

After some consideration, the 2015 IEEE Signal Processing Cup data set was selected for this project. The motives for its selection were the fact that it is seen as the standard when dealing with this subject, being used in most papers, as well as the fact that it is one of the few to provide a ground truth for the heart rate, which could be essential for this project.

5.1.1 Overall Structure

This dataset is divided into three main folders: "competition_data", "TestData" and "True_BPM", as seen in figure 5.1.
The first of these folders contains a two ".mat" files for each of its 12 subjects, with one file contained the recorded PPG, ECG and acceleration data and the other containing the ground truth meaning the heart rate of the subject in every eight second long time window extracted from the chest mounted ECG.
The second of those folders contains a single ".mat" file for each of its eight subjects, with the corresponding ground truth for each of them being stored in the "True_BPM" folder. However one thing that should be noted is that though these files have a similar structured to the ones mentioned before, they do not contain any ECG data.

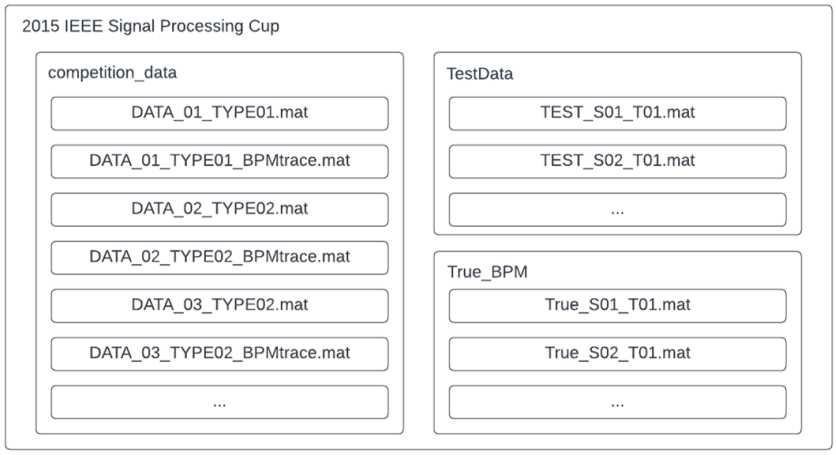

Figure 5.1: Representation of the structure of the 2015 IEEE Signal Processing Cup database.

5.1.2 Signals

All 3 signals, ECG, PPG and acceleration, were sampled at 125Hz and collected from subjects withing the 18-35 age range. When it came to the recording of the PPG signal of each of them, the dataset actually contains two recordings, from two pulse oximeters, one placed in each wrist. These oximeters were actually part of a wrist-worn device that also included the accelerometer which was used to obtain the acceleration signal. Finally the ECG was recorded from a chest mounted device.

Regarding the structure of the actual dataset files: both the files with the names of type "Data_XX_TYPEXX" and type "TEST_SXX_TXX" have a similar structure, with both containing a variable named 'sig' that has six rows in the case of the first type, as represented in figure 5.2, and five rows in the case of the second. The extra row in the "competition_data" files represents the ECG signal and is the first row of each file. The following five rows are the same as in the "TestData" files, first two rows for each of the PPG channels and then three rows, representing the acceleration data, one for each axis, X, Y and Z.

	0	1	2	3	...	40798	40799	40800	40801	40802	
0	-262.5000	-264.0000	-261.0000	-260.5000	...	-369.0000	-389.0000	-371.0000	-359.5000	-367.5000	ECG Signal
1	-13.5000	-15.0000	-15.0000	-16.0000	...	191.5000	206.0000	218.0000	224.5000	227.0000	PPG Signal - Right Wrist
2	-2.5000	-3.5000	-2.5000	-2.5000	...	162.5000	177.0000	190.0000	199.5000	205.0000	PPG Signal - Left Wrist
3	0.0702	0.0624	0.0624	0.0702	...	0.1092	0.1170	0.1014	0.1014	0.0858	Acceleration Signal - X Axis
4	0.1950	0.1872	0.1950	0.1950	...	0.7020	0.6942	0.7098	0.7098	0.7098	Acceleration Signal - Y Axis
5	0.8268	0.8346	0.8268	0.8268	...	0.0468	0.0390	0.0546	0.0702	0.0858	Acceleration Signal - Z Axis

Figure 5.2: Representation of the "sig" variable Dataframe from the 2015 IEEE Signal Processing Cup database.

Regarding the naming of the each file, it should be noted that all files have one number referring to the subject and then a type number which is referent to the kind of exercise they were put trough upon the recording of the signals.

When it comes to the "competition_data", the type number can be "01" or "02", both reffering to an exercise in which the subjects ran on a treadmill, differing in the sequence of different speeds, as observed in figure 5.3:

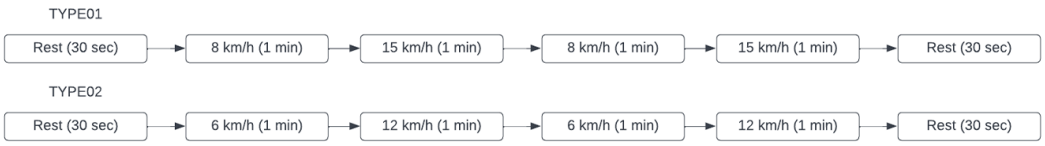

Figure 5.3: Representation of the types of exercises in the "competition_data" from the 2015 IEEE Signal Processing Cup database.

When it comes to the "TestData", the type number can be "01" or "02", with the first referring to activities like shaking hands, stretching, and other kinds of arm movements that can be seen in rehabilitation exercises, while the second involves more intensive arm exercises like boxing.

5.1.3 Ground truth

When it comes to the ground truth, each one of the values corresponds to an heart rate estimation for each eight second time window of every signal. However, these windows overlap, with only two seconds of two contiguous time windows being original, for example: If the first ground truth is refferent to a time window that starts on the first sampling point and ends at sampling point number 1000 (since at a 125Hz refresh rate, the 1000 sampling points is equivalent to eight seconds), the following window will go from the 251th sampling point to the 1250th sampling point. It should also be noted that the ground truth is saved in a different file, with the ground truths of the files in the "competition_data" being saved in files of name type "DATA_XX_TYPEXX_BPMtrace", while the ones refferent to the "TestData" files are saved as "True_SXX_TXX".

5.1.4 Restructuring of the data

When it comes to restructuring the data from the dataset so that it is suited to our model, two different approaches were taken due to the nature of this paper:

5.1.4.1 PPG-based structure

First, we needed a data structure where only the PPG data would be used, so a data frame was built in which each row was referent to one 8 second window, with each column being a sampling point, except for the last one which would actually contain the ground_truth referent to that time window, as shown in figure 5.4.

sampling points

	0	1	2	3	4	5	...	997	998	999	Ground_Truth
0	-23.0	-24.0	-26.5	-27.0	-30.0	-30.0	...	33.5	33.0	31.5	74.339207
1	-20.5	-19.0	-20.0	-19.5	-20.5	-18.5	...	15.0	-14.5	-13.5	76.357466
2	22.0	23.5	22.5	23.5	22.5	21.5	...	15.5	14.0	14.5	77.142857
3	-5.5	-5.5	-6.5	-6.5	-7.0	-7.0	...	11.0	11.5	10.5	74.668142
4	30.5	27.0	26.0	24.0	21.5	20.5	...	18.5	-17.5	-17.5	72.580645
...	...	...	...	...	...	...	...	...	...	...	...
1761	130.0	142.5	148.0	155.0	156.5	153.5	...	83.5	79.0	72.0	159.066800
1762	-33.5	-36.0	-38.0	-38.5	-36.5	-33.5	...	47.0	-56.5	-63.5	157.397700
1763	-38.0	-41.5	-43.5	-44.5	-38.5	-31.5	...	89.0	-92.0	-93.0	156.250000
1764	111.5	100.5	85.5	70.0	53.5	36.5	...	29.0	-24.5	-20.0	155.440400
1765	66.5	59.5	51.5	44.0	36.0	30.5	...	9.5	18.5	28.0	154.004100

time windows

Figure 5.4: Representation of the dataframe referent to the PPG-based model.

5.1.4.2 Acceleration-based structure

To test the possibility that the inclusion of acceleration data into the structure could lead to a better notion of the motion artifacts that affect the PPG data and therefore more accurate heart rate predictions, we must also build another dataframe that takes into account this data. This one had to be a bit more complex, taking a three dimensional shape, with the first axis being the number of time windows, the second being the length of each of those time windows and the final one being the number of signals taken into account, as shown in figure 5.5.

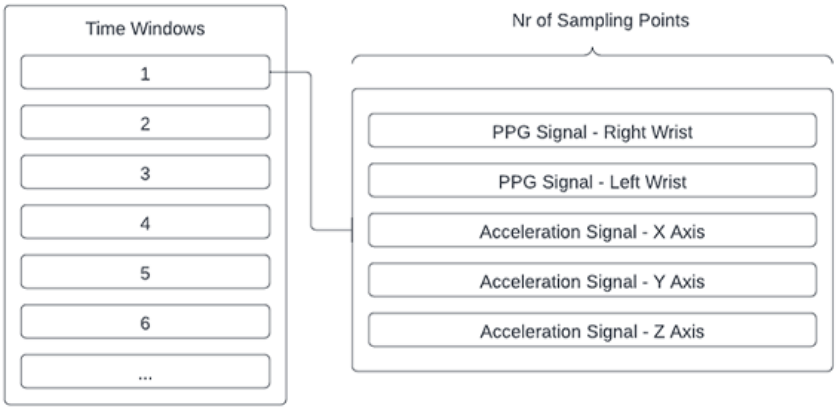

Figure 5.5: Representation of the dataframe referent to the Acceleration-based model.

5.2 Deep Learning Models

When it came to creating the models to run the data on, two main ones were developed, first a more simplified one based on the most common implementations of CNN and LSTM layers and then a more complex one derived from the examples found during the literature review, that was expected to be more well suited to the data and therefore provide better results.

5.2.1 Simplified Model

The first model used was referred to as the simplified model and it was based upon several implementations of CNN and LSTM based models that can be found online and across many papers, such as [21]. As described in figure 5.6, this model contains only four layers:

- A one-dimensional Convolutional layer, with its main hyperparamaters that would eventually be tuned being the size of its kernel and the number of filters.
- A Long Short Term Memmory layer whose tunable hyperparamters were solely its number of units and with an Hyperbolic tangent activation function.
- A Flatten layer which transforms the output of the LSTM layer, so that it becomes one dimensional.
- Finally a dense layer, with a single unit.

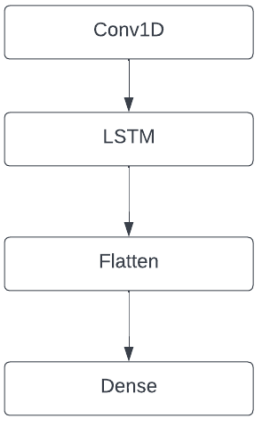

Figure 5.6: Representation of the simplied model's layer structure.

5.2.2 Final Model

The second model used was referred to as the final model and it was based upon the implementations found across the many papers present in the literature review, mainly [2]. As shown in figure 5.7, this model contains 15 layers:

- Two one-dimensional Convolutional layers, just like in the previous model, accompanied by four other layers each:
  - A Batch Normalization function, with the goal of normalizing the output to a mean of zero and a standard deviation of one.
  - An Activation layer, defining its activation function as a rectified linear unit.
  - A one dimensional max-pooling layer with the goal of downsampling the output and with one tunable hyperparameter, its pool size.
  - A Dropout layer with the goal of preventing overfitting by resetting the weights, with one single tunable hyperparameter, the rate of dropout.
- Two Long Short Term Memmory layers whose tunable hyperparamters were solely their number of units and both with an Hyperbolic tangent activation function. Their only difference being that the first was set to return sequential data while the second was set to return the last output of the output sequence.
- A Flatten layer which transforms the output of the LSTM layer, so that it becomes one dimensional.
- Finally two dense layers, one with multiple units and the other with a single unit, divided by a single Flatten layer. The only tunable hyperparameter being the number on units on the first of those layers.

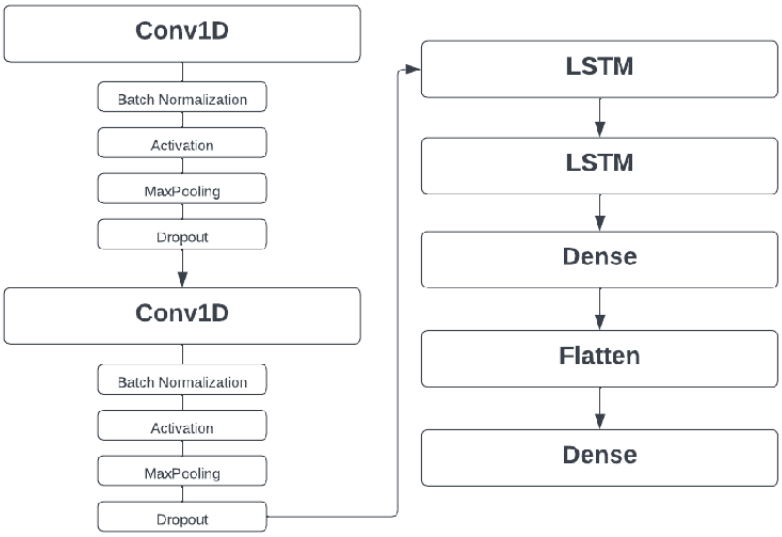

Figure 5.7: Representation of the final model's layer structure.

5.2.3 Optimizers

The same optimizer was used across all models, this being Root Mean Squared Propagation or RMSprop. Though other optimizers like the commonly used, Adam optimizer were tested, this one always seemed to provide the best results. RMSProp was created with the intent of making the optimization process faster using techniques like Gradient Descent.

5.2.4 Tuning

For tuning, KerasTuner was the framework used, most often utilizing Random Search across a range of values seen as appropriate for each tunable hyperparameter mentioned previously. The Random Search was always stopped at around 65 trials, which has been shown to be close to being as effective as Grid search.

5.2.5 Performance Evaluation

For performance evaluation we used multiple metrics, as a way to get a better, more in depth look at the nature of the predictions, given the massive size of the data , such as Mean Average Error, Root Mean Squared Error, as well as R2 score, which is described as "the proportion of the variance in the dependent variable that is predictable from the independent variables", as well as the Standard Deviation. Their selection was loosely based on the array of metrics provided in [2]. Alongside this, the validation loss and loss of the models was also plotted to facilitate the detection of certain phenomenons, like overfitting.

5.3 Experiment Design

In this section the steps taken for each experiment included in the scope of this project will be thoroughly explained starting with an overall look at the project workflow and then a look at specific aspects of this project that are worthy of a more thorough explanation.

5.3.1 Overall Workflow

Once the main dataset used in this project had been selected, then the first methods to be implemented were the traditional ones like peak detection and some other strategies with spectrograms. After this was all done, it was time to begin implementing some deep learning methods, and so, using the concepts learned at the previous stage, an initial structure for the data was created. Alongside this a first model was created with barely any layers, as well as methods to determine the quality of the results.
From here on out, these last few steps were repeated in a cycle, as shown in figure 5.8 always improving upon the last iteration of every step with the goal of getting the cleanest and highest quality results.

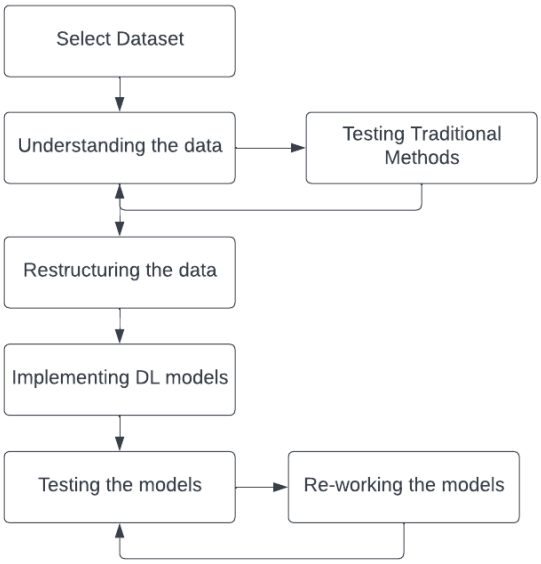

Figure 5.8: General representation of the workflow of the project.

For this, one main point of interest that stood out was the split used for the data. With the dataset that was selected being quite heterogeneous in its nature, given the different kinds of activities employed by the subjects during the gathering of the data.
Initially, a random train test split was used but that rapidly became obsolete once it was noted that is could cause the spill of data from the same subject or even the same specific sampling points (due to the overlapping time windows) between the training and testing sets. Here, many strategies seen in the literature were employed, mostly revolving around many alternative cross validation techniques, such as leave-one-window-out validation, where in each iteration of a cycle that goes trough the entire dataset, a window is selected for testing, while every other is selected for training, with the overlapping windows being disregarded. However this method was not only computationally infeasible given the size of this dataset but it also did not account for the spill of same-subject data between the training and testings sets. From this moment one a derivation of this method refereed to as leave-one-subject out validation was selected, employing the exact same strategy but using the data from each subject as the unit to operate upon, rather than each single window. Eventually culminating in the creation of an array of different scenarios, from which the most relevant ones were selected to be discussed in this paper, those being the many combinations of the different data structures and models mentioned above. However, alongside this many tangents began surfacing that demanded more attention and these were given some time as well, as they seem as relevant to this paper as the main objective.

5.3.2 Window Size

One of the other experiments that was run focused on the size of the window from which the ground truth was obtained. As mentioned before, the ground truth that was made available by the

dataset represented the heart rate of each subject at every eight second time window, with each pair of consecutive windows overlapping by six seconds. However, while working on this project, the size of that window began be put into questions, being that it was one of the few variables that, given its nature, had never been tampered with. To start there were a few attempts at extrapolating ground truth values for windows of different sizes from the ones provided by the dataset, but that proved to be sub-optimal as there was always the possibility that the ground truth values obtained were inaccurate. This led to the search for an alternative and given the previous work with peak detection and the fact that the dataset's documentation mentions that the ground truth had been obtained from the ECG data in it, the question was raised of whether peak detection could be used on the ECG data in order to generate new ground truth data over different window sizes that could be used to run the deep learning models. This experiment proved fruitful, with the results obtained from using the ground truth from the peak detection over the same 8 second long time window proved to have similar results to what had been previously obtained with the default ground truth provided by the dataset. From this moment on many scenarios were run, with the time windows of sizes four seconds, eight seconds, 16 seconds, 32 seconds and one minute being selected for the scope of this project.

5.3.3 Validation Dataset

Finally, towards the final stages of the project, a separate dataset with a matching structure to the one that had previously been used, named PPG-DaLiA was selected in order to run further testing, ensuring that the hyperparameters of the created models had not been over tuned to the point that they were unable to get high results on foreign data.

5.4 Summary

After selecting the 2015 SCP dataset and tracing its structure, the shortcomings of the raw data were noted and improved upon, restructuring it in a way so that it would be suitable for the models that were bound to be created. Then two models were created the baseline model and the final model, which differed in complexity. These were created over a continuous cycle of work with the aim of improving the results, which were defined trough four metrics, MAE, RMSE, R2 score and Standard Variation.

Chapter 6

Results and Discussion

This chapter comprises and discusses all of the results obtained along the development of this project that were deemed as relevant, looking at how they may or may not support the hypothesis brought up within the scope of this dissertation, as well as other key aspects that could raise questions that may lead to future work or that may have led to changes along the workflow of this one.

6.1 Baseline Model Performance

For the baseline model, only one scenario was selected, where the PPG signals from both wrists were used alongside the 3 available axis of acceleration data and the results were the following:

	MAE (bpm)	RMSE (bpm)	R2 Score	STD (bpm)
competition_data	10.236 bpm	12.611 bpm	0.494	7.325 bpm
only activity set nr2	9.492 bpm	12.335 bpm	0.594	7.829 bpm
Test Data	14.538 bpm	17.091 bpm	-1.786	8.745 bpm
only activity set nr3	7.149 bpm	8.709 bpm	0.080	4.954 bpm
only activity set nr4	13.627 bpm	16.674 bpm	-1.058	9.290 bpm
Entire Dataset	11.359 bpm	14.061 bpm	-0.324	8.127 bpm

Table 6.1: Table describing the results obtained when running the baseline model on PPG and Acceleration data.

This scenario was kept in the workflow and deemed relevant as it served as a way to ensure the effectiveness of the suggested final model, a sort of reference point for bench marking the results it was bringing.

The results obtained in table 6.1, validate the expectation that most CNN-LSTM model architectures should bring about the same sort of result structure that was previously observed.

The most notable piece of information that can be highlighted in table 6.1, is perhaps the fact that the activity set number 3 still presented relatively decent results in this scenario, indicating that perhaps the final model could be tuned so that it was more suited to it, as all the other scenarios

seem to display a higher degree of improvement in their results when compared to this one.

For this experiment the hyper parameters used were the following:

Nr of filters in first convolution layer	64
Kernel size of first convolution layer	62
Number of units in first LSTM layer	64

Table 6.2: Table describing the best hyperparameters used in the experiment where the baseline model was used.

6.2 Final Models Performance

For the final model, two different scenarios were selected, one where only the data obtained trough the PPG was used and another where the acceleration data that was made available alongside it in the dataset is also taken into account.

6.2.1 Using only PPG data

In this scenario the only data used was the PPG signal from the right wrist of the subject, and the results were the following:

	MAE (bpm)	RMSE (bpm)	R2 Score	STD (bpm)
competition_data	5.025 bpm	7.416 bpm	0.734	5.337 bpm
only activity set nr2	5.432 bpm	7.499 bpm	0.606	5.128 bpm
Test Data	13.711 bpm	16.239 bpm	-1.472	8.639 bpm
only activity set nr3	6.027 bpm	7.511 bpm	0.319	4.463 bpm
only activity set nr4	10.978 bpm	13.044 bpm	-0.126	6.742 bpm
Entire Dataset	8.772 bpm	11.856 bpm	-0.269	7.847 bpm

Table 6.3: Table describing the results obtained when running the final model on only PPG data.

While analyzing the results in table 6.3 the first thing that becomes evident is the substantially lower quality results obtained when running the model exclusively on the activity set number 4, this could be attributed to the nature of the kinds of exercises done in this set (e.g. boxing) generating much more complex motion artifacts than those on the other sets (e.g. running on a treadmill), which more than often generate motion artifacts that are generally cyclical, as the subject is forced to repeat the same motions and is given a lesser degree of freedom for his movements. The negative R2 score is one major negative sign, as it means the predictions are not following the trend of the data, as seen in figure 6.1, displaying a low correlation which is most likely due to the model struggling to deal with the heavy noise.

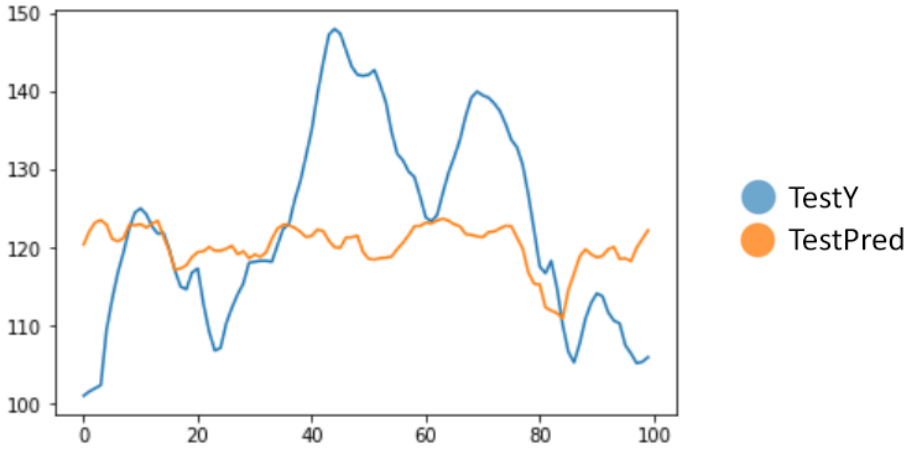

Figure 6.1: Representation of the results obtained while using only activity set number 4, while only using the ppg signal as input for the model, displaying the predictions inability to follow the trend of the data. (blue line represents the real values, orange line represents the predictions made by the model).

However, the mean absolute errors between five to 6 in the other scenarios are hopeful, especially in the scenarios where only the competition data is being used, with much higher R2 scores, reaching up to 0.7, meaning that not only is the model making accurate predictions, but also precise ones that are representative of the real trends observed in the data.

For this experiment the hyper parameters used were the following:

Nr of filters in first convolution layer	64
Kernel size of first convolution layer	20
Pool size of first max pooling layer	8
Dropout rate of first dropout layer	0.1
Nr of filters in second convolution layer	32
Kernel size of second convolution layer	20
Pool size of second max pooling layer	8
Dropout rate of second dropout layer	0.6
Number of units in first LSTM layer	32
Number of units in second LSTM layer	128
Number of units in first Dense layer	16

Table 6.4: Table describing the best hyperparameters used in the experiment where only ppg was used on the final model.

6.2.2 Pairing PPG data with Acceleration data

In this scenario the PPG signal from both wrists was used as well as all three axis of acceleration data, and the results were the following:

	MAE (bpm)	RMSE (bpm)	R2 Score	STD (bpm)
competition_data	3.957 bpm	5.906 bpm	0.819	4.272 bpm
only activity set nr2	4.014 bpm	5.565 bpm	0.828	3.789 bpm
Test Data	8.507 bpm	10.151 bpm	-0.388	5.374 bpm
only activity set nr3	5.214 bpm	6.539 bpm	0.540	3.922 bpm
only activity set nr4	9.501 bpm	11.769 bpm	0.028	6.543 bpm
Entire Dataset	7.033 bpm	9.058 bpm	0.137	5.376 bpm

Table 6.5: Table describing the results obtained when running the final model on PPG and Acceleration data.

When it comes to the results in table 6.4, it can be observed that the results follow a similar structure to the one observed in table 6.3, except for the fact that they are all of significantly higher quality.

However there are few details that are especially interesting. First of all, that the R2 score observed when the entire dataset was used is positive this time around, confirming that to some extent the predictions are following the trend of the data, which can be seen as proof of the main hypothesis of this dissertation, meaning that the model is being able to use the acceleration data to produce better predictions from the same noisy data. This could come of a result of the model being able to detect when sampling points are being more or less affected by the motion artifacts, giving them a higher or lower weight when it comes to making the predictions. This is supported by the fact that when it comes to the heart rate, consecutive sampling points or time Windows tend to have approximate heart rates. This positive trend in the quality of the R2 scores obtained in this experiment can also be observed when using only the test data or only the data from activity set number 4. In fact it is in these two scenarios that the R2 score was subject to the most prominent increases, as seen in figure 6.2. Given the fact these scenarios also represent the activity sets that

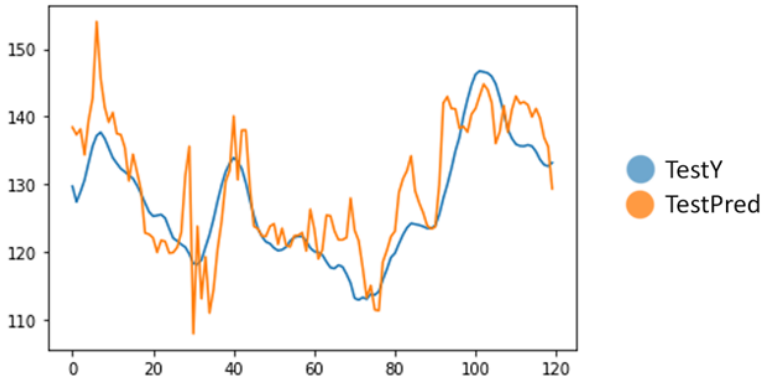

Figure 6.2: Representation of the results obtained while using only activity set number 4, while using acceleration data alongside the ppg signal, displaying the predictions improved ability to follow the trend of the data. (blue line represents the real values, orange line represents the predictions made by the model).

tend to produce the most noise, could be seen as justification to the statement that the addition of

acceleration data to the data that was fed to the model improved the results by allowing the model to better understand what sampling points were more affected by the motion artifacts, rather than the acceleration data having any direct correlation to the heart rate.

The same can be argued over the fact that the difference in the quality between the results obtained in the scenarios where only competition data or only test data was used was much lower now.

For this experiment the hyper parameters used were the following:

Nr of filters in first convolution layer	32
Kernel size of first convolution layer	30
Pool size of first max pooling layer	8
Dropout rate of first dropout layer	0.3
Nr of filters in second convolution layer	64
Kernel size of second convolution layer	15
Pool size of second max pooling layer	8
Dropout rate of second dropout layer	0.1
Number of units in first LSTM layer	32
Number of units in second LSTM layer	64
Number of units in first Dense layer	50

Table 6.6: Table describing the best hyperparameters used in the experiment where both ppg and acceleration data were ran on the final model.

6.3 Other Experiments

6.3.1 Window Size

In this experiment, 5 different windows sizes were selected, including the regular window size of 8 seconds even though, in this scenario, its labels were also obtained trough ECG peak detection, and the results were the following:

	MAE (bpm)	RMSE (bpm)	R2 Score	STD (bpm)
4 seconds	11.176 bpm	14.097 bpm	0.431	8.548 bpm
8 seconds	8.291 bpm	10.633 bpm	0.553	6.657 bpm
16 seconds	6.964 bpm	8.675 bpm	0.661	5.086 bpm
32 seconds	6.380 bpm	8.343 bpm	0.656	5.347 bpm
60 seconds	5.019 bpm	6.320 bpm	0.673	3.780 bpm

Table 6.7: Table describing the results obtained from multiple executions of the final model with ground truths obtained via peak detection of the ECG trough multiple window sizes.

As mentioned previously, trough out the workflow, one main question that was frequently raised was whether or not the size of the window from which the ground truth was obtained had any effect in the results which was why this experiment was created. From the further inspection

of the results in 6.7, it can be noted that there is a clear tendency for the results to improve as the time window is increased. Although initially there was some discussion over whether or not this was to be expected, the consensus was that it made sense that with more information available to the model at each time window, this was to be expected. Still, even though this could be easily seen as a simple way to go about getting higher quality results, if this model were to be used in a real life scenario the time window would be inversely proportional to the frequency at which the heart rate of the subject would be able to be displayed, meaning that a time window of 1 minute would render its usage to be futile. This leads to another question, which is the most suitable time window dimensions for real life usage which, considering the results would most like be some place between 8 and 16 seconds, has the rate of improvement seems to greatly decline after this point, with a time window of 32 seconds not offering even a full unit of improvement over the results obtained with a time window of 16 seconds.

6.3.2 Validation Dataset

Finally, the results obtained when running the final model with PPG and Acceleration data from the PPG-DaLiA dataset (as well as previous results obtained with the 2015 SCP dataset, for comparison) were as represented in table 6.8:

	MAE (bpm)	RMSE (bpm)	R2 Score	STD (bpm)
PPG-only	8.772 bpm	11.856 bpm	-0.269	7.847 bpm
PPG and Acceleration	7.033 bpm	9.058 bpm	0.137	5.376 bpm
PPG-DaLiA Dataset	9.520 bpm	12.742 bpm	0.370	8.443 bpm

Table 6.8: Table describing the results obtained when running the model on a different dataset.

This final experiment seeked only to ensure that the hyperparameters that were in use were not over tuned. The results that came out of this experiment were largely positive, with the mean absolute error only slightly increasing when using this dataset. However there were some key points that deserve some attention: First of all, the R2 score was actually higher in this dataset, which shows that in this experiment the model was more capable of producing predictions that followed the trend of the data, this can be attributed to the aforementioned heterogenity of the activities employed by the subject in the 2015 SCP dataset, as in the PPG-DaLiA dataset all the subjects are suggested to the same activity set, even if it may incorporate many different activities in itself.

6.4 Summary

In general, these results were heavily favourable to the hypothesis that were raised at the beggining of this dissertation, it can clearly be seen that acceleration data can be greatly helpful when it comes to obtaining higher quality results from the use of a PPG device, even and especially during intense physical activity. The biggest drawback or negative point to raise is that the model seems to

have a hard time dealing with widely varying ranges of different motion ranges at the same time. And of course, it was very interesting to note the effects that different sized time windows can have in the results, something that so far has not been widely observed or debated in the existing literature and that the model seems to have positive results running on other datasets of the same nature, proving we have managed to avoid overly tuning to the specific circumstances brought to the table by this one.

Chapter 7

Conclusion and Future Work

7.1 Conclusions

The clear conclusion that can be taken from this dissertation is, of course, that CNN-LSTM models can be used and are a promising medium for the accurate prediction of the heart rate of a subject that is performing intense physical exercise. Alongside this there is also the conclusion that the usage of acceleration data and its pairing with the PPG data can be greatly beneficial to the quality of the results.

Out of all the proposed models, the final model, when using both PPG and acceleration data as its input, was the one to obtain the highest quality results, with a mean absolute error of just 3.957 ± 4.272 bpm when running solely on the competition data provenient from the 2015 SCP dataset and 7.033 ± 5.376 bpm when running on the entire dataset. The vadility of the model was ensured by running it on the PPG-DaLiA dataset, in which it was able to reach a mean absolute error of 9.520 ± 8.443 bpm, a result deemed relevant and indicative of its validity.

Trough these experiments, the decision to explore further testing with different window sizes was also reached, leading us to the conclusion that increasing window sizes lead to higher quality results, though it was noted that higher window sizes could also have drawbacks in real life applications of the ppg technology.

It can be said that this dissertation demonstrated potential when it comes to different applications of deep learning, more specifically CNN-LSTM, models in the field of heartrate prediction. The implementation of these models in real life scenarios could facilitate the attainment of accurate metrics that could be used in the diagnosal of episodic and sporadic heart disease, such as arrhythmia, as it could remove the need for the use of cumbersome and stationary machines that stop the subject from being tested in real life scenarios that are conducive to organic motion.

7.2 Future Work

Even though it has been declared that the developed work has achieve its initial goals, it has also exposed many scenarios and opportunities for its further development.

First of all, it exposed the limitations that are currently imposed by the available data sets. Following the same train of thought, it would also be interesting, if more data is made available, to explore other more streamlined and robust methods for the determination of ground truth values across different window sizes trough the ECG data.
It would also be an interesting opportunity to explore the introduction of a pre-processing stage of denoising of the data, perhaps trough operations such as quadratic smoothing. In fact, the pre-processing stage is something that is still open to some work, as the stage of exploring statiscal and more traditional methods of heart rate prediction as the literature review was in flow, provided great insight into many interesting alternative that are available. It could be said that the encapsulation of the different types of available methods and approaches, as well as their combination, could be the key to creating innovative pipelines capable of producing more accurate results, taking advantage of the strengths of each of them.
Finally, it can, of course, be noted that the exploration of the multitude of other network designs that were outside of the reach of this dissertation due to its time constraints, such as Transformers, could also be a pathway to new and interesting discoveries.

References

[1] Shikder Shafiul Bashar, Md Sazal Miah, AHM Zadidul Karim, Md Abdullah Al Mahmud, and Zahid Hasan. A machine learning approach for heart rate estimation from ppg signal using random forest regression algorithm. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–5. IEEE, 2019.

[2] Dwaipayan Biswas, Luke Everson, Muqing Liu, Madhuri Panwar, Bram-Ernst Verhoef, Shrishail Patki, Chris H Kim, Amit Acharyya, Chris Van Hoof, Mario Konijnenburg, et al. Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment. *IEEE transactions on biomedical circuits and systems*, 13(2):282–291, 2019.

[3] Rikiya Yamashita et al. Convolutional neural networks: An overview and application in radiology. *Insights into Imaging*, 9(4):611–29, 2018.

[4] Gunther Eysenbach. What is e-health? *Journal of medical Internet research*, 3(2):e20, 2001.

[5] Gartheeban Ganeshapillai and John Guttag. Real time reconstruction of quasiperiodic multi parameter physiological signals. *EURASIP Journal on Advances in Signal Processing*, 2012(1):1–15, 2012.

[6] David J Gladstone, Melanie Spring, Paul Dorian, Val Panzov, Kevin E Thorpe, Judith Hall, Haris Vaid, Martin O'Donnell, Andreas Laupacis, Robert Côté, et al. Atrial fibrillation in patients with cryptogenic stroke. *N engl J med*, 370:2467–2477, 2014.

[7] Shahid Ismail, Usman Akram, and Imran Siddiqi. Heart rate tracking in photoplethysmography signals affected by motion artifacts: a review. *EURASIP Journal on Advances in Signal Processing*, 2021(1):1–27, 2021.

[8] Anand Nayyar Jafar Alzubi and Akshi Kumar. Machine learning from theory to algorithms: An overview. *Journal of Physics: Conference Series 1142*, 2018.

[9] Gu-Young Jeong, Kee-Ho Yu, and Nam-Gyun Kim. Continuous blood pressure monitoring using pulse wave transit time. *:* , pages 834–837, 2005.

[10] Stefan C. Kremer John F. Kolen. Gradient flow in recurrent nets: The difficulty of learning longterm dependencies. *A Field Guide to Dynamical Recurrent Networks*, 2009.

[11] Alex Pappachen James Kamilya Smagulova. A survey on lstm memristive neural network architectures and applications. *The European Physical Journal Special Topics*, 228(10):2313–24, 2019.

[12] Samy Bengio Dumitru Erhan Oriol Vinyals, Alexander Toshev. Show and tell: A neural image caption generator. 2015.

[13] Tania Pereira, Cheng Ding, Kais Gadhoumi, Nate Tran, Rene A Colorado, Karl Meisel, and Xiao Hu. Deep learning approaches for plethysmography signal quality assessment in the presence of atrial fibrillation. *Physiological measurement*, 40(12):125002, 2019.

[14] Tania Pereira, Kais Gadhoumi, Mitchell Ma, Xiuyun Liu, Ran Xiao, Rene A Colorado, Kevin J Keenan, Karl Meisel, and Xiao Hu. A supervised approach to robust photoplethysmography quality assessment. *IEEE Journal of Biomedical and Health Informatics*, 24(3):649–657, 2019.

[15] Lukasz Piwek, David A Ellis, Sally Andrews, and Adam Joinson. The rise of consumer health wearables: promises and barriers. *PLoS medicine*, 13(2):e1001953, 2016.

[16] K.C. Gupta Qi-Jun Zhang and V.K. Devabhaktuni. *Artificial Neural Networks for RF and Microwave Design - from Theory to Practice*. 2003.

[17] A Shyam, Vignesh Ravichandran, SP Preejith, Jayaraj Joseph, and Mohanasankar Sivaprakasam. Ppgnet: Deep network for device independent heart rate estimation from photoplethysmogram. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1899–1902. IEEE, 2019.

[18] Andriy Temko. Accurate heart rate monitoring during physical exercises using ppg. *IEEE Transactions on Biomedical Engineering*, 64(9):2016–2024, 2017.

[19] Yoshua Bengio Yann LeCun and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–44, 2015.

[20] Ali K Yetisen, Juan Leonardo Martinez-Hurtado, Barış Ünal, Ali Khademhosseini, and Haider Butt. Wearables in medicine. *Advanced Materials*, 30(33):1706910, 2018.

[21] J. Zhang and S. Li. Air quality index forecast in beijing based on cnn-lstm multi-model. *Chemosphere*, 308.

[22] Qingxue Zhang, Xuan Zeng, Wenchuang Hu, and Dian Zhou. A machine learning-empowered system for long-term motion-tolerant wearable monitoring of blood pressure and heart rate with ear-ecg/ppg. *IEEE Access*, 5:10547–10561, 2017.

[23] Zhilin Zhang, Zhouyue Pi, and Benyuan Liu. Troika: A general framework for heart rate monitoring using wrist-type photoplethysmographic signals during intensive physical exercise. *IEEE Transactions on biomedical engineering*, 62(2):522–531, 2014.