A Thesis Submitted for the Degree of PhD at the University of Warwick

Permanent WRAP URL:

http://wrap.warwick.ac.uk/151189

Copyright and reuse:
This thesis is made available online and is protected by original copyright.
Please scroll down to view the document itself.
Please refer to the repository record for this item for information to help you to cite it.
Our policy information is available from the repository home page.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk
Deep Learning for Electrocardiogram Analysis
Applied to Health Monitoring Applications

by

Mihaela Porumb

Thesis
Submitted to the University of Warwick
for the degree of
Doctor of Philosophy

School of Engineering
February 2020
Contents

List of Tables vi
List of Figures viii
Acknowledgments xvii
Declarations xviii
Abstract xx
Abbreviations xxi

Chapter 1 Introduction 1
  1.1 Chapter overview 1
  1.2 Introduction to the research 1
  1.3 Research questions, aim and objectives 5
  1.4 Overview of the research method 8
  1.5 Structure of the thesis 11

Chapter 2 Methodology background 13
  2.1 Chapter overview 13
  2.2 Background on time series and sequences 13
  2.3 Background on Time Series Classification (TSC) 14
      2.3.1 Traditional, data mining approaches for TSC 15
      2.3.2 Deep Learning (DL) for sequence representation learning 16
  2.4 Modules of DL used for time series data 18
      2.4.1 Multilayer Perceptron (MLP) 19
          2.4.1.1 Forward pass 20
          2.4.1.2 Output layer 20
          2.4.1.3 Objective function 20
<table>
<thead>
<tr>
<th>2.4.1.4 Backward pass</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.4.2 Convolutional Neural Network (CNN)</td>
<td>22</td>
</tr>
<tr>
<td>2.4.2.1 Visualising what is going on inside a CNN</td>
<td>24</td>
</tr>
<tr>
<td>2.4.2.2 Visualising which input subsequences are class discriminative</td>
<td>25</td>
</tr>
<tr>
<td>2.4.3 Recurrent Neural Network (RNN)</td>
<td>26</td>
</tr>
<tr>
<td>2.4.4 Autoencoder (AE)</td>
<td>30</td>
</tr>
<tr>
<td>2.4.4.1 Denoising Autoencoder</td>
<td>31</td>
</tr>
<tr>
<td>2.4.4.2 Convolutional Autoencoder</td>
<td>31</td>
</tr>
<tr>
<td>2.5 Embeddings visualisation</td>
<td>33</td>
</tr>
<tr>
<td>2.6 DL networks training</td>
<td>34</td>
</tr>
<tr>
<td>2.6.1 Optimisation algorithms</td>
<td>34</td>
</tr>
<tr>
<td>2.6.2 Generalisation</td>
<td>36</td>
</tr>
<tr>
<td>2.6.2.1 Early stopping</td>
<td>36</td>
</tr>
<tr>
<td>2.6.2.2 Regularisation</td>
<td>37</td>
</tr>
<tr>
<td>2.6.2.3 Other methods for improving generalisation</td>
<td>38</td>
</tr>
<tr>
<td>2.6.3 Input representation</td>
<td>38</td>
</tr>
<tr>
<td>2.6.4 Weight initialisation</td>
<td>39</td>
</tr>
<tr>
<td>2.6.5 Hyperparameters</td>
<td>41</td>
</tr>
<tr>
<td>2.7 Conclusions</td>
<td>42</td>
</tr>
</tbody>
</table>

**Chapter 3** Literature review on DL for Electrocardiogram (ECG) analysis

<table>
<thead>
<tr>
<th>3.1 Chapter overview</th>
<th>43</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2 ECG fundamentals</td>
<td>43</td>
</tr>
<tr>
<td>3.2.1 ECG anatomy and physiology</td>
<td>43</td>
</tr>
<tr>
<td>3.2.2 Acquisition</td>
<td>44</td>
</tr>
<tr>
<td>3.2.3 ECG intra- and inter-subject variability</td>
<td>45</td>
</tr>
<tr>
<td>3.3 Review of ECG analysis applications with DL</td>
<td>46</td>
</tr>
<tr>
<td>3.3.1 Literature search</td>
<td>47</td>
</tr>
<tr>
<td>3.3.2 Inclusion and exclusion criteria</td>
<td>47</td>
</tr>
<tr>
<td>3.3.3 Paper selection and data extraction</td>
<td>47</td>
</tr>
<tr>
<td>3.3.4 Analysis of data extracted on DL for the ECG analysis</td>
<td>48</td>
</tr>
<tr>
<td>3.3.5 Results</td>
<td>49</td>
</tr>
<tr>
<td>3.3.5.1 Public ECG data.</td>
<td>49</td>
</tr>
<tr>
<td>3.3.5.2 Bibliometric analysis</td>
<td>52</td>
</tr>
<tr>
<td>3.3.5.3 Recurrent DL architectures and their structure</td>
<td>54</td>
</tr>
</tbody>
</table>
Chapter 4 Deep Learning for Low Glucose Detection using ECG Recordings

4.1 Chapter overview

4.2 Background on glucose levels prediction

4.3 Case study 1 - Free-living conditions
  4.3.1 Experiment protocol
  4.3.1.1 Ethical approval
  4.3.2 Dataset
  4.3.2.1 CNN based system dataset
  4.3.2.2 CNN + RNN based system dataset
  4.3.3 Data pre-processing
  4.3.4 Proposed DL architecture
    4.3.4.1 CNN architecture
    4.3.4.2 CNN + RNN architecture
  4.3.5 Performance evaluation
  4.3.6 Localisation of the contributing ECG heartbeat subsequences
  4.3.7 Visualisation of the data in a lower dimensional space
  4.3.8 Statistical analysis
  4.3.9 Programming
  4.3.10 Results
    4.3.10.1 Classification of ECG signals that correspond to normal/low glucose values
    4.3.10.2 CNN based system
    4.3.10.3 CNN+RNN based system
    4.3.10.4 Localisation of the discriminative heartbeat subsequences
    4.3.10.5 Dimensionality reduction – CNN based system
    4.3.10.6 Statistical analysis
  4.3.11 Discussion
  4.3.12 Case study 1 - Conclusions

4.4 Case study 2 - Human Metabolism Research Unit (HMRU) conditions
Chapter 5  Deep Learning for Congestive Heart Failure Diagnosis using ECG Recordings  
5.1 Chapter overview .................................................. 124
5.2 Background on Heart Failure (HF) assessment ................. 125
5.3 Dataset ................................................................. 127
5.4 Data pre-processing .................................................. 128
5.5 Heartbeat classification ............................................. 128
  5.5.1 Performance measures ......................................... 130
5.6 Proposed CNN architecture ....................................... 130
  5.6.1 Visualisation of class specific sequences .................. 131
  5.6.2 Experimental setting ......................................... 132
5.7 Results .................................................................. 133
  5.7.1 Individual heartbeat classification ......................... 133
  5.7.2 Classification results using the majority voting strategy ... 134
  5.7.3 DL model motivation ........................................... 135
  5.7.4 Class-specific sequences ...................................... 137
5.8 Discussion .............................................................. 139
5.9 Conclusions ............................................................ 143

Chapter 6  Deep Learning for Heart Rate Variability (HRV) Features Estimation from Raw ECG Recordings  
6.1 Chapter overview .................................................... 144
6.2 Background on HRV computation ................................ 145
6.3 Dataset ................................................................. 148
6.4 Data pre-processing .................................................. 149
6.5 Proposed DL architecture .......................................... 150
## 6.5.1 Architecture choice motivation ........................................ 152
## 6.5.2 Experimental setting ..................................................... 153
## 6.6 Results ............................................................................. 153
## 6.7 Discussion ........................................................................ 154
## 6.8 Conclusions ...................................................................... 156

### Chapter 7 Conclusions and Future Work 157

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td></td>
</tr>
<tr>
<td>7.3</td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td></td>
</tr>
</tbody>
</table>

### References 167

### Appendix A Supplementary materials for Chapter 3 193

### Appendix B Supplementary materials for Chapter 4 206

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>B.1</td>
<td></td>
</tr>
<tr>
<td>B.2</td>
<td></td>
</tr>
</tbody>
</table>
List of Tables

3.1 Summary table of the major ECG databases used in the surveyed publications ........................................... 50
3.2 Summary of the surveyed articles by the employed DL method ............................................................... 58

4.1 Participants’ demographics, eligible participants highlighted in green ......................................................... 73
4.2 Recommended target blood glucose levels by NICE and ADA ........................................................................ 75
4.3 Number of extracted heartbeats from the nighttime ECG recordings for each eligible participant enrolled in the study and their distribution in the training and testing datasets .................................................. 80
4.4 Total number of extracted heartbeats from the nighttime ECG recordings for each eligible participant and the final number of heartbeats selected for the analysis ........................................................................................................ 80
4.5 Total number of extracted 5-minute ECG excerpts from the nighttime ECG recordings for each eligible participant enrolled in the study and their distribution in the training and testing datasets. ........................................................................ 81
4.6 CNN based model classification results, evaluated for each participant. The individual beat column presents the results for each heartbeat classification, whereas the 10-minute voting column shows the results when taking the majority class corresponding to all heartbeats in a 10-minute window of time. .................................................. 89
4.7 CNN+RNN classification results, evaluated for each participant. The 5-minute column presents the results for the 5-minute ECG segments used during training, whereas the 10-minute voting column shows the results when taking the majority class corresponding to all heartbeats in a 10-minute window of time. .................................................. 91
4.8 Kruskal-Wallis H-test between pairs of subjects for the extracted ECG features corresponding to Low glucose (a) and Normal glucose (b). *p-value post hoc* column presents the post hoc pairwise test results for multiple comparisons of mean rank sums using Dunn’s test.

4.9 Classification results on the test dataset using 2D, 20D and 50D embedding sizes in the CDAE and 100 optimisation steps of the CDAE. The reported values represent % / 100.

4.10 Classification results on the validation dataset using 2D, 20D and 50D embedding sizes in the CDAE and 100 optimisation steps of the CDAE. The reported values represent % / 100.

4.11 Recent hypoglycemia detection studies based on ECG analysis results

5.1 Total number of extracted heartbeats from both datasets:

5.2 Average per-heartbeat classification results as a mean of 10 runs on 10 random splits of the input data into training/validation/testing datasets. Results presented as % / 100.

5.3 Mean confusion matrix for the 10 different runs

5.4 Test subjects with misclassified 5-minute ECG segments

5.5 Classification performance of the classifier proposed in the current study and those from other relevant previous studies

6.1 Common time domain HRV measures, taken from [364]

6.2 Common frequency domain HRV measures, taken from [364]

6.3 5-minute HRV features available for each of the selected 500 subjects enrolled in the SHHS study

6.4 Total number of 5-minutes intervals considered in training, validation and testing datasets

6.5 Test dataset Long Short-Term Memory (LSTM) regression results corresponding to four HRV features: AVNN, SDNN, HF and IHR for 250 time steps (thus, equivalent of 250 input heartbeats)

6.6 Test dataset - LSTM results obtained for AVNN and IHR features considering different number of time steps

6.7 Number of correct predictions for AVNN and IHR using a 5% and 10% error threshold
List of Figures

2.1 Learning time series representations (based on Madalina Fiterau’s talk ‘Learning Representations from Time Series Data through Contextualized LSTMs’, presented in Neural Information Processing Systems Conference (NIPS), 2016) [68] ................. 18
2.2 MLP example .................................................. 21
2.3 Generic CNN network structure, comprises of input layer, several convolutional, pooling and fully-connected layers. Figure generated using the online tool NN-SVG, available at http://alexlenail.me/NN-SVG/AlexNet.html. .................................................. 23
2.4 Example of 1-D convolutions of an input with size = 7 and a 1-D filter with size = 3. The example on the left illustrates a convolution with a stride of 1 whereas the example of the right a convolution using a stride of 2, with no padding in both cases. The filter weights are shown on the very right and have the values [1, 0, -1] and bias 0. (based on [76]) .................................................. 24
2.5 A simple RNN, unrolled across time steps, based on [82] ............. 27
2.6 LSTM memory cell, based on [82] ......................... 28
2.7 An RNN with a hidden layer consisting of 2 memory cells, unfolded across time, based on [80]. .................................................. 30
2.8 General denoising AE diagram. The input example $x$ is corrupted to $\tilde{x}$, the AE then maps $\tilde{x}$ to $y$ and tries to reconstruct $x$. .................. 32
2.9 Example of the CDAE. The CDAE is composed of convolutional and deconvolutional (convolutional transpose) layers and the latent representation($y$). The input for the CDAE is a corrupted version($\tilde{x}$) of $x$ that is used as an output. Usually, the learned parameters from training the CDAE are used to initialise a classifier for different target classes. .................................................. 32
2.10 Overfitting on training data. The error on the validation dataset initially decreases, however after several training epochs it reaches a plateau and starts to increase. The dashed line indicates the ideal point on the best performance on the validation dataset, therefore, the early stopping point can be used to stop the training process.

3.1 The sequence of depolarisation and repolarisation of the heart, related to the waves in the ECG tracing. Image marked as public domain.

3.2 Flow chart presenting the results of the literature search.

3.3 Bibliographic coupling map of the selected publications with respect to the referenced documents.

3.4 Bibliographic coupling map of the selected publications with respect to the referenced sources.

3.5 Histogram of the number of publications identified through keyword search per year of publication. The darker bars represent the number of publications selected using the inclusion/exclusion criteria defined above.

3.6 Histogram of DL applications using ECG signal, presented in the selected publications using the inclusion/exclusion criteria. For the articles that addressed multiple applications, each application is considered in the histogram.

3.7 Author’s keywords co-occurrence in selected papers, included in the review as distance-based map. Colours indicate the cluster to which a keyword was assigned, also the distance between two items reflects the strength of the relation between them.

3.8 Author’s keywords co-occurrence in the selected papers included in the review, presented as a density map. The colour of a point in the map depends on the number of items in the neighbourhood of the point and on the importance (weights) of the neighbouring items.

3.9 Histogram of the surveyed articles per DL architectures. The acronyms for the architectures are generic, therefore they can denote various sub-types/customisations of the base DL architecture. For the articles that employed multiple DL methods, each method was separately considered in the histogram.
3.10 Stacked histogram of the surveyed articles per application and the employed DL architectures. The CNN and AE architecture variations were collapsed to CNN and AE for enhancing the visualisation. For the articles that employed multiple DL methods and applications, each method and application was separately considered in the histogram.

3.11 Heatmap of the CNNs’ architecture structure employed in the reviewed studies. The x axis represents the number of conv layers, each row in the heatmap represents the number of filters (left) and the sizes of the filters (right) in each reviewed CNN architecture. For the publications that employed more than one architecture, each CNN architecture was considered, therefore the total number of reviewed architectures is higher (107) than the total number of reported studies that employed CNN based methods (65). (first part)

3.11 cont.

3.12 Correlations and kernel density estimate plots for inspecting the relation between the input size versus the number of convolutional layers. The low correlation coefficient ($r = 0.14$) and the p value $> 0.01$ confirms that there is no linear correlation between the two variables.

4.1 Flow chart of the experimental protocol, including data collection, data preprocessing and model building.

4.2 Colour maps of the extracted heartbeats during a 24h period (figure on the left) for each participant (figures a, b, c, d) associated with their corresponding glucose and activity levels (figure on the right).

4.3 Nightime (midnight to 9 AM) glucose histograms corresponding to all the available participants. The glucose value of the $10^{th}$ percentile corresponding to participants 5, 6, 7 and 8 is higher (4.9, 7.1, 5.1 and 4.7 mmol/L) than the expected values of 4.2 mmol/L, thus 4 participants could not be considered for further analysis.
4.4 Proposed CNN based system illustrating the study objectives. To detect the low glucose levels using the ECG signal, three objectives were set: (OBJ 1) was to build a classifier (using a CNN network) for the low glucose levels detection task. Secondly, the chosen method for performing the classification (i.e. CNN) enables us to investigate further the learned representation of the input heartbeats (OBJ 2), representation (embedding) that can be used in for data visualisation/clustering in lower-dimensional space. The method used for the nonlinear dimension reduction is t-Distributed Stochastic Neighbor Embedding (t-SNE) [92]. The third objective (OBJ 3) was to investigate the important regions in the input time series (the heartbeat signal) that contribute the most to the final classification result (Gradient Class Activation Map (Grad-CAM) method). . . . . . . . . . . 83

4.5 CNN architecture hyperparameters search . . . . . . . . . . . . . . . 83

4.6 Proposed CNN + RNN system for low blood glucose detection over a 5-minute window of time. The individual heartbeats were firstly isolated, then grouped into 5-minute segments. Each considered 5-minute segment was chosen if it contained at least 200 heartbeats. This condition also implies that the glucose event (low/normal) should last for at least 5 minutes, thus each 5-minute ECG segment was associated with a single label: low/normal glucose. Each heartbeat was firstly transformed into a feature representation using a CNN network, a representation that was fed as input to the sequence model (LSTM cells). The outputs of the final RNN are the inputs to a linear layer with a softmax producing a distribution P over the two possible outputs: normal or low glucose values. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 85
4.7 Hypoglycemia detection during the night using the heartbeat majority voting in the 10-minute window of time. The black waveform represents the glucose values recorded by the Continuous Glucose Monitor (CGM), considered as ground truth glucose level in this study. The grey shaded regions illustrate a ±10% error boundary for the CGM glucose readings [267], [268] as reported in previous studies. The colour of the points indicates the predicted class: red for the predicted low glucose levels and green for the predicted normal glucose levels. The dark colours indicate more certain predictions: dark red points accounted for low glucose predictions with the predicted probability $>0.7$, while light red accounted for low glucose prediction with predicted probability $\leq 0.7$; dark green accounted for normal glucose prediction with predicted probability $>0.7$ and light green accounted for normal-glucose prediction with a probability $\leq 0.7$. Images (a)-(h) present the glucose level predictions for a sample training day, and columns (i)-(p) present the glucose predictions for a sample test day.

4.9 t-SNE visualisation of the test heartbeats corresponding to subject 3 in the activation space representation. The red heartbeats correspond to a low glucose level ($< 4.0$ mmol/L) and the green heartbeats correspond to normal glucose levels. a. t-SNE visualisation when the heartbeats are coloured according to the glucose annotation (class) b. t-SNE visualisation using a colour map that shows the glucose value associated with each heartbeat, the darker the colour the higher the glucose value for normal beats and the lower for low glucose beats.
4.8 Identification of the most relevant heartbeat segments for hypoglycemia detection using the Grad-CAM method. The solid lines represent the mean of all the heartbeats that correspond to each subject in the training dataset: green during normal glucose levels, red during hypoglycemic events. The comparison among 4 different subjects highlighted the fact that each subject may have a different ECG waveform during hypoglycemic events, for instance Subjects 1 and 2 present a visibly longer QT interval during hypoglycemic events, different from subjects 3 and 4. The error bands represent the standard deviation of the considered heartbeats. The vertical bars represent the histograms of the sample points that were > 0.9 in the normalised heatmaps obtained from applying Grad-CAM methods on all the training heartbeats.

4.10 Box plots for the extracted ECG features during Low and Normal glucose levels for every participant. A multi-way Kruskal-Wallis H-test was performed for every ECG parameter for the low and normal glucose condition separately. The only non-significant differences between the groups are indicated in the plot n.s.

4.11 Point plots for the extracted ECG features during Low and Normal glucose levels for every participant, showing the relationship between the mean of every ECG feature for low and normal glucose levels.

4.12 Mann-Whitney rank test on the extracted ECG parameters for each subject, figures a-d. All the statistical tests showed significant differences between the groups (low vs normal glucose level).

4.13 Selection of the study participants

4.14 Structure of the CDAE and classifier for the low glucose detection task. The CDAE is composed of convolutional and deconvolutional (convolutional transpose) layers and the latent representation (y), which contains either 20 or 50 neurons, is used for both classification and clustering(t-SNE). The input for the t-SNE is a corrupted version(\tilde{x}) of a heartbeat(x) that is used as output. The learned parameters from training the CDAE are used to initialise a classifier whose outputs are the glucose levels(low/normal).

4.15 Example of reconstructed heartbeats for different embedding sizes, for two of the participants.
4.16 MSE and MAE for the output heartbeat reconstruction on the test dataset for all the study participants. The boxen plots show the MSE and MAE for different embedding sizes (2D, 20D and 50D), and for 100, 500, and 1000 CDAE optimisation iterations.

4.17 Visualisation of the learned embeddings for the participant with ID=6. For 20D and 50D the t-SNE method was employed to reduce the input dimensionality of the embedding to 3D. t-SNE was optimised for 5000 steps and the perplexity parameter was set to 25.

4.18 CDAE iterations classification results on validation and testing datasets

4.19 Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. Red points correspond to predicted low glucose level whereas the green point represent the normal glucose predictions. Lighter color intensities (red or green) denote a lower probability of the prediction $\leq 0.7$.

5.1 Overview of the heart failure assessment studies, similar to [333]

5.2 Sample 10s ECG corresponding to a Congestive Heart Failure (CHF) (top) patient and a healthy subject (bottom). The selected heartbeats illustrate the 5s selection in the pre-processing step.

5.3 Proposed CNN architecture for beat classification and visualisation of the class discriminative sequences in the input time series.

5.4 Heartbeats classification for the subjects included in the training, validation and test datasets, evaluated on one of the 10 runs. Panels b) and d) present the predictions for the voting in a 5-minute ECG window. The red and green bars represent the extracted heartbeats in chronological order for CHF patients and healthy subjects, respectively. The grey gaps in the middle of the bars represent missing heartbeats.

5.5 Example of misclassified ECG segments evaluated on the validation and testing datasets.

5.6 Loss during training.

5.7 Histograms of the gradients of the last convolutional layer, for each training step. It can be observed that the neurons are not completely saturated at 0, and that the gradients are preserving a normal distribution with zero mean.
5.8 Histograms of the activations in the last convolutional layer, for each training step. This plot shows that both the Xavier weight initialisation and the batch normalisation are preventing the neurons from being saturated.

5.9 Histograms obtained from the Grad-CAM heat maps, indicating the most important sample points in the input beats for a certain class. Each bar in the histograms represents a single sample point in the heartbeat time series, thus the greater the bar the more important that point is for the classification result.

6.1 Electrophysiology of the heart, schematic diagram of a normal sinus rhythm in (a) and Heart Rate Variability computed from the RR series in (b). Image marked as public domain.

6.2 Flow chart summarising the steps for ECG processing for computing HRV, similar to [364].

6.3 Distribution of the 5-minute segments availability for the 500 subjects.

6.4 Histogram presenting the distribution of the number of heartbeats in the 5-minute ECG intervals.

6.5 Flow chart summarising the ECG preprocessing steps.

6.6 Proposed RNN architecture.

6.7 Distribution plot of the error between computed and predicted AVNN (a) and IHR (b).

6.8 Bland Altman plot for the AVNN (a) and IHR (b) obtained on the test dataset.

A.0 Table presenting details regarding the DL architectures and preprocessing steps for the 106 shortlisted papers included in the review.

B.1 Subject 1. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.

B.2 Subject 2. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.

B.3 Subject 3. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.

B.4 Subject 4. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.

B.5 Subject 1. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
B.6 Subject 2. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . . . . . . . . . 219

B.7 Subject 3. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . . . . . . . . . 220

B.8 Subject 4. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . . . . . . . . . 221

B.9 Subject 5. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . . . . . . . . . 222

B.10 Subject 6. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . . . . . . . . . 223

B.11 Subject 7 and Subject 8. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. . 224
Acknowledgments

My utmost gratitude goes to my supervisor, Dr. Leandro Pecchia, who gave me the opportunity to be part of the Applied Biomedical Signal Processing and Intelligent eHealth Lab (ABSPIE) research group at the University of Warwick. Leandro taught me how to do research, he gave me the freedom and the support to pursue my own ideas, while providing me essential guidance towards achieving the research goals. I would also like to thank my second supervisor, Dr. John Hattersley for support and guidance throughout my PhD and also for enabling the data collection in the Human Metabolism Research Unit at the University Hospitals Coventry and Warwickshire.

I want to send my gratitude to the former and current research students and postdocs that I met at Warwick during my PhD, for making the Biomedical Engineering lab such a fun and pleasant working environment.

Thank you to my parents for their continuous support and encouragements throughout this journey.

Lastly, but not least, special thanks to my husband Octav for being very supportive when I needed the most, for his optimism, reassurance and confidence that helped me to pursue my dream.
Declarations

This thesis is submitted to the University of Warwick in support of my application for the degree of Doctor of Philosophy. It has been composed by myself and has not been submitted in any previous application for any degree.

The work presented (including the data analysis) was carried out by the author except in the case studies outlined below:

- The work presented in chapter 3 is a secondary analysis, a review of previously published data. These data were extracted from 106 original articles, which are cited in this thesis.
- The work presented in chapter 5 was performed on a public dataset retrieved from MIT-BIH Normal Sinus Rhythm Database [1] and BIDMC Congestive Heart Failure Database [2] included in PhysioNet [3].
- The work presented in chapter 6 is a secondary analysis of a public dataset retrieved from Sleep Heart Health Study (SHHS) [4]–[7].

Parts of this thesis have been published or submitted for publication by the author:

*Journal article, published online*


xviii
Journal article, submitted


Patent application


Conference articles


Abstract

Several recent advances fuelled the significant increase in interest for Artificial Intelligence-based healthcare innovations: the vast availability of biomedical data, accurate wearable sensors, electronic health records, the advancement in Machine Learning methods and affordable computational resources. This thesis focuses on electrocardiogram signal analysis using a range of deep learning techniques, including Convolutional Neural Networks, Recurrent Neural Networks and Convolutional Autoencoders for developing several health monitoring applications. The performance of the proposed models was investigated in two applications (i.e. nocturnal low glucose detection and congestive heart failure diagnosis), that share the same aspects of the input data including noise, time-dependence, heterogeneity. This thesis explores the efficacy of a personalised deep learning-based system for the automatic detection of low glucose levels in real-life settings, overcoming the limitations of previous attempts. Furthermore, this thesis explores unsupervised methods for learning time series representations to address the high cost and scarcity of the labelled biomedical data. A novel deep learning-based method that employed raw electrocardiogram signals was explored for congestive heart failure diagnosis. Added to their superior performance, the results presented in this thesis bring forward the intuition that short electrocardiogram recordings, of just about 5 minutes, can be sufficient to diagnose correctly severe congestive heart failure. A third case study presented in this thesis advances the idea that certain heart rate variability features can be estimated from a short (≈ 1 minute) raw electrocardiogram signal, which might facilitate the development of real-time applications.

This thesis shows that leveraging deep learning-based models for physiological signal analysis, not only bypasses the costly feature extraction and selection process, but they can reveal unintuitive patterns in the input data that are class-discriminative. Providing transparency on how the models reached certain conclusions is crucial for the healthcare field, firstly because the medical professionals expect specific evidence for their recommendations and secondly, the models can help doctors better understand the physiological processes.

Overall, the findings of this thesis confirmed that deep learning methods applied for electrocardiogram analysis could improve and extend current diagnostic models, might bring new research opportunities and contribute to the development of real-time health monitoring applications.
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-NN</td>
<td>One Nearest Neighbour.</td>
</tr>
<tr>
<td>AE</td>
<td>Autoencoder.</td>
</tr>
<tr>
<td>AI</td>
<td>Artificial Intelligence.</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial Neural Network.</td>
</tr>
<tr>
<td>AUC</td>
<td>Area Under the Curve.</td>
</tr>
<tr>
<td>BW</td>
<td>Baseline Wander.</td>
</tr>
<tr>
<td>CAE</td>
<td>Convolutional Autoencoder.</td>
</tr>
<tr>
<td>CAM</td>
<td>Class Activation Map.</td>
</tr>
<tr>
<td>CART</td>
<td>Classification and Regression Trees.</td>
</tr>
<tr>
<td>CDAE</td>
<td>Convolutional Denoising Autoencoder.</td>
</tr>
<tr>
<td>CGM</td>
<td>Continuous Glucose Monitor.</td>
</tr>
<tr>
<td>CHF</td>
<td>Congestive Heart Failure.</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional Neural Network.</td>
</tr>
<tr>
<td>CRNN</td>
<td>Convolutional Recurrent Neural Network.</td>
</tr>
<tr>
<td>DBN</td>
<td>Deep Belief Network.</td>
</tr>
<tr>
<td>DL</td>
<td>Deep Learning.</td>
</tr>
<tr>
<td>DTW</td>
<td>Dynamic Time Warping.</td>
</tr>
<tr>
<td>DWT</td>
<td>Discrete Wavelet Transform.</td>
</tr>
<tr>
<td>ECG</td>
<td>Electrocardiogram.</td>
</tr>
<tr>
<td>FNN</td>
<td>Feedforward Neural Network.</td>
</tr>
<tr>
<td>GAN</td>
<td>Generative Adversarial Network.</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>GAP</td>
<td>Global Average Pooling.</td>
</tr>
<tr>
<td>Grad-CAM</td>
<td>Gradient Class Activation Map.</td>
</tr>
<tr>
<td>GRU</td>
<td>Gated Recurrent Unit.</td>
</tr>
<tr>
<td>HF</td>
<td>Heart Failure.</td>
</tr>
<tr>
<td>HMRU</td>
<td>Human Metabolism Research Unit.</td>
</tr>
<tr>
<td>HR</td>
<td>Heart Rate.</td>
</tr>
<tr>
<td>HRV</td>
<td>Heart Rate Variability.</td>
</tr>
<tr>
<td>k-NN</td>
<td>k-Nearest Neighbours.</td>
</tr>
<tr>
<td>LSTM</td>
<td>Long Short-Term Memory.</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning.</td>
</tr>
<tr>
<td>MLP</td>
<td>Multilayer Perceptron.</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Square Error.</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis.</td>
</tr>
<tr>
<td>PLI</td>
<td>Power Line Interference.</td>
</tr>
<tr>
<td>ReLU</td>
<td>Rectified Linear Unit.</td>
</tr>
<tr>
<td>RNN</td>
<td>Recurrent Neural Network.</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine.</td>
</tr>
<tr>
<td>t-SNE</td>
<td>t-Distributed Stochastic Neighbor Embedding.</td>
</tr>
<tr>
<td>TSC</td>
<td>Time Series Classification.</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Chapter overview

This chapter introduces the research aspects. Explanation of the research topic (section 1.2), the aim and objectives (section 1.3) and an overview of the research method (section 1.4) are presented in this chapter. Finally, a detailed structure of the thesis is presented in section (1.5).

1.2 Introduction to the research

Artificial Intelligence (AI) promises to revolutionise our society through its ability to account for complex interactions to identify patterns from raw data. Healthcare is not an exception to this trend; medicine is considered as one of the fast growing application domains in AI, with substantial potential socioeconomic impact. Several recent advances fuelled the exponential increase in interest for AI-based healthcare innovations: the vast availability of biomedical data, the accurate wearable sensors that can continuously monitor all vital signs, the integrated databases and electronic health records, the advancement in Machine Learning (ML) methods and open-source ML libraries and the affordable and powerful computational resources and cloud computing.

Emerging biomedical data sources include health records, wearable and sensor data, imaging, -omics, which share common features: they are complex, high-dimensional, usually noisy and unstructured [11]. Therefore, the characteristics of the biomedical data bring unprecedented opportunities and at the same time, significant challenges to healthcare research. Extracting relevant information from complex, multi-modal data to explore the associations between different variables...
to diagnose diseases, determine the appropriate treatment, assist physicians in complex decisions or generate diagnostic hypothesis is required for the successful development of clinical decision support systems, as emphasised in recent reviews on AI in healthcare [12], [13].

In the early 1970s attempts to develop such decision support systems leveraged rule-based approaches, which initially showed promising results [14]. However, with the increase in data availability and complexity of the problems to be solved, rule-based systems proved to be costly to build and fragile, as they heavily relied on expert input, on explicitly devised formal rules and human-designed updates [12].

Contrary to the initial rule-based systems which were heavily dependent on domain experts and robust rule development, recent ML methods have provided more advanced methods for developing predictive tools, by leveraging statistical algorithms that allow computers to learn directly from data. Therefore, ML methods have been widely applied to healthcare data, the potential applications are vast, including screening disease detection and classification, patient risk stratification, disease prediction and treatment selection. ML is a field of AI, based on computational and statistical algorithms that allow computers to acquire their own knowledge, by extracting patterns from the input data. A formal, succinct definition for a machine learning algorithm was given by Mitchell in 1997: 'A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.' Based on this definition, common tasks in terms of how the machine learning algorithms should process an input example include classification, classification with missing inputs, regression, machine translation, structured output learning, anomaly detection, data synthesis, imputation of missing values, denoising, density estimation and many other [15]. The most widely used performance measure is accuracy; however, it is a task-specific measure. Lastly, machine learning algorithms can be divided into two categories: supervised and unsupervised, based on the allowed experience during the learning process. Supervised learning algorithms experience a dataset that contains examples that also have associated target values. Simple ML algorithms work by learning a mapping from an input representation they are given to the desired output target. Therefore, in order for a ML system to provide useful predictions, they do not examine the patient directly. Instead, they are provided with a series of information, also known as features [15]. Thus, it becomes clear that the performance of the ML algorithms depend substantially on the representation of data they are given, as the ML algorithms cannot influence the structure or the content of the features in any way.
However, many AI tasks proved they could be solved by designing the right set of features, then presenting these features to a ML algorithm. A few successful examples include acute mental stress detection [16], Congestive Heart Failure (CHF) detection and severity classification [17], prediction of cardiovascular and cerebrovascular events [18], classification of the antibodies associated with renal graft failure [19], identification of type 2 diabetes [20], prediction of drugs effects and interactions [21] and many others.

As emphasised by Goodfellow et al. [15], for many tasks, it is difficult if not impossible, to know what features should be extracted. One solution to this problem would be to use AI methods to discover the right representations for the task automatically, therefore to determine the AI-based methods to learn not only the mapping from the input representation to the output but also the representation itself [15]. These approaches are known as representation learning; in the past few years it has been shown that automatically learned representations most often result in better performance than hand-designed ones and can easily adapt to new tasks with minimal human interventions [15].

Briefly, Deep Learning (DL) is a type of ML algorithm that provides a solution for the representation learning problem by introducing hierarchical representations that are expressed in terms of simpler representations. DL methods were initially inspired by how the brain works, consisting of linked nodes, also called neurons, arranged in a sequence of layers, so that neurons get activated according to the input they receive. The high representational power of DL algorithms (hereafter called models or modules) is given by the layer-wise arrangement of the neurons, enabling the computer to build complex concepts based on simpler ones.

DL is a relatively new field and has become the state of the art method for several applications, especially in the computer vision field, greatly advancing the performance of visual recognition systems [22]. The prevalence and success of DL methods are due to their ability to automatically extract relevant features from raw data. This property greatly reduces the need for heavy preprocessing of the input data, bypasses the common feature extraction and feature selection steps, which usually require a great human effort, they are time-consuming, might involve the domain experts for the design and most of the time the features need to be redesigned in an iterative process for every new task.

Given its demonstrated performance in different domains such as speech recognition, translation systems and computer vision, DL methods came to the attention of the biomedical research community. Even large scale technology companies such as Google DeepMind and Enlitic announced plans to apply DL techniques
to healthcare back in 2016 [23], [24]. In healthcare DL techniques were mainly used for the interpretation of medical images, such as retinal fundus images for diagnosing age-related macular degeneration and diabetic retinopathy [25], or skin images to classify lesions as benign lesions or malignant skin cancers [26].

Many aspects of DL such as end-to-end learning from raw data, ability to handle complex, multi-modal data, and integrated feature learning could be useful for addressing some of the challenges of biomedical data: sparsity, noise, time-dependent, heterogeneous. However, much of the focus has been on applications of DL methods for static data and less on time series data.

A representative example of a physiological time series is the Electrocardiogram (ECG) signal, a widely used recording for clinical screening which captures the cardiac electrical activity from the body surface. ECG signals are mainly used for diagnosing cardiovascular disorders, which according to the World Health Organization are responsible for 30% of deaths worldwide. ML, and in particular DL techniques, have recently been applied to address the analysis of ECG signals, chapter 3 provides a comprehensive review on the recent DL applications for the analysis of ECG signals.

Learning models for real-life time series data, such as ECG, introduces additional challenges due to the temporal component of the data, i.e. individual data points cannot be assumed to be independent and identically distributed (i.i.d). Furthermore, the sensors used to record the data can introduce a series of measurement errors due to noise, faults, sensor calibration, drift, temperature and humidity dependent upon the types used. Therefore, common approaches for handling real-life time series data involve heavy preprocessing of the input signal by removing noise, reducing complexity and extracting relevant features. Another traditional approach for modelling sequential data is to estimate the parameters from an inferred model of the data and then use the estimated parameters as informative features. However, noisy real-life time series data are difficult to describe with equations since the dynamics of the system are most of the time unknown or too complex to be fully captured with a relatively small number of non-linear operations. The alternative is to use DL for the automatic feature representation learning of time series data.

In this thesis, the focus was on the application of DL models for sequence classification and regression applied to public ECG data and to ECG data that were acquired with wearable sensors. Moreover, the need for learning time series representations from unlabelled data is even more pronounced in the medical domain as the labelling process requires the intervention and supervision of domain experts. To address this problem, this thesis investigates the feasibility of construct-
Another advantage of employing certain DL models for time series classification is the possibility to localise the class-specific regions in the input data. Localising the important regions in the input space is of particular interest for medical applications, this might help clinicians to understand better and explain the underlying physiological processes. In this thesis, several known methods in the computer vision field of visualising class-specific regions are adapted and employed for time series data. Another challenge when dealing with sequential input is the integration of other structured covariates in the model. In healthcare applications, it is prevalent that sequential data is accompanied by structured information, thus a method for integrating other variables in a DL model is also investigated.

The primary contribution of this research is the application of DL to real-life time series data (i.e. ECG recordings), for the diagnosis and detection of two different clinical conditions: the detection of low glucose levels (aka. hypoglycemia) and the diagnosis of severe CHF. The proposed approach in this research for the detection of low glucose levels is a novel, personalised, approach that has not been explored in the past and that has shown promising results, presented in detail in chapter 4. A secondary contribution of this thesis is the investigation of time series feature learning from unlabelled data in order to boost the performance of a time series classification system and to enable training of the DL models with less available data. Another contribution of this research is the adjustment of a general method for visualisation of class-specific sequences to time series input data, a method employed in both case studies: CHF and hypoglycemia detection. A final contribution of the thesis is the application of DL methods on a benchmark dataset and advance the idea that extracting certain Heart Rate Variability (HRV) features for building prediction models can be bypassed by employing DL models on raw ECG data.

Although the focus of the thesis is on biomedical applications, the proposed frameworks are generalisable and can work with any time series input data that have fixed length and the same sampling frequency.

1.3 Research questions, aim and objectives

The recent progress and innovation in the accuracy and availability of wearable sensors (e.g. smartwatches that can record the heart rate and estimate an activity level, smartphones, inertial sensors, ECG wearable sensors) has opened new research possibilities and encouraged the development of new methods and systems for real-
life health monitoring. The possibility of recording and tracking of continuous, real-time physiological signals opened unprecedented opportunities and increased the demand for developing methods and tools that can handle an extensive amount of data in real-time, sometimes multi-dimensional noisy, signals. These characteristics of physiological data have recently determined a greater interest in the application of DL methodologies for physiological signal analysis rather than employing more traditional signal processing techniques.

However, real-life settings condition pose great challenges and introduce several limitations for signal processing and analysis. Outside of a controlled environment, users are free to carry out their daily activities, which alter less the physiological response, but at the same time, a large amount of noisier data are generated. Thus, the use of wearable devices in real-life settings has raised new challenges for both the sensor development and data analysis communities. Moreover, the adoption of these wearable devices enabled the exploration of novel health monitoring applications. Therefore, real-life signal recordings and DL based methods enabled both an improvement in the currently existing models for prediction of certain conditions and at the same time have the potential to revolutionise continuous health monitoring systems. This research aims to explore different DL-based methods for ECG analysis to develop several health monitoring applications. In particular, the research in this thesis explored and investigated the following questions:

**Research Question 1:** How appropriate are DL-based methodologies for physiological data analysis, in particular for ECG analysis:
- is it feasible to use DL methods for automatic time series representation learning, for dimensionality reduction and extraction of relevant information?
- is it feasible to apply DL methods for physiological time series classification and regression?

**Research Question 2:** By leveraging DL methods, is it feasible to develop a personalised, non-invasive, non-intrusive low glucose detection system based on raw ECG signals analysis, recorded with wearable devices? Would an unsupervised feature learning step improve the performance of the low glucose detection system?

**Research Question 3:** Using DL based methods, is it feasible to analyse raw ECG data for the diagnosis of CHF? Are DL based methods more accurate than traditional feature-based methods?

**Research Question 4:** Using DL methods is it possible to infer certain 5-minutes HRV features from a raw, shorter ECG excerpt?
**Research Question 5:** Is it possible to improve transparency and interpretability of DL models for time series analysis? In particular, is it possible to determine the essential subsequences in the ECG heartbeat that are used by the DL models for reaching specific outcomes?

Accordingly, the main objectives of this research are:

**Objective 1:** To review the main ECG signal-based applications that employed DL methods, identifying the optimal DL architectures, ECG preprocessing steps, and model validation strategies for specific tasks.

**Objective 2:** To develop a proof-of-concept for a personalised, non-invasive, ECG-based low glucose detection system using wearable devices, in free-living conditions using the representational power of DL techniques.

**Objective 3:** To investigate whether using an unsupervised step for pretraining the DL model improves the performance of the low blood glucose detection system.

**Objective 4:** To investigate the performance of DL models for CHF diagnosis on a benchmark dataset and to compare their performance with traditional feature-based models.

**Objective 5:** To determine whether HRV features can be inferred from a short (less than 1 minute), raw ECG excerpt by leveraging DL techniques.

**Objective 6:** To devise and validate a general method for visualisation of class-discriminative subsequences in the input time series, thus to enhance the model transparency when employing specific DL modules.

In order to fulfill these objectives, the following case studies were designed and performed in this research:

**Study 1:** A detailed analysis of the recent publications (including 106 studies) on DL employed for ECG, published in the past 3 years, was performed extracting and analysing information regarding bibliometrics, DL architectures, preprocessing steps.

**Study 2:** A prospective study performed in free-living conditions on 8 healthy participants that underwent 14-days continuous monitoring of glucose levels and
ECG using wearable sensors for the development of a proof-of-concept system for low glucose detection in real-life settings.

**Study 3:** A prospective study performed on 25 healthy, over 60s that underwent a 12-week exercise and nutrition intervention on healthy ageing, were continuously monitored during a 24 or 36 hours visit at University Hospitals Coventry & Warwickshire, in a calorimeter room, having recorded continuous glucose levels and ECG signals during their hospital visit. The data collected from these healthy participants was used to provide further evidence supporting the feasibility of a non-invasive, ECG-based system for low glucose detection and to investigate certain unsupervised methods for improving the performance of the DL models.

**Study 4:** A secondary study using a public dataset comprising of long-term ambulatory ECG recordings corresponding to both healthy subjects and individuals diagnosed with severe CHF (33 subjects in total, ≈ 20 hours of ECG/subject) was used to develop an automatic CHF diagnostics system using DL methods.

**Study 5:** A secondary study using a public dataset comprising of 500 subjects with high quality ECG recordings (≈ 7 hours of ECG/subject) was used to determine whether certain HRV features computed on a 5-minute ECG excerpt can be inferred from a short (≈ 1 min) ECG from raw ECG recordings.

The fulfillment of these objectives required the application of a broad and diverse set of research methodologies, tools and techniques for the collection and analysis of data, which are briefly described in section 1.4.

### 1.4 Overview of the research method

In study 1 (chapter 3), recent ECG applications and the proposed DL methods for ECG analysis were investigated, extracting information about the recurrent DL models, their structure with respect to the input size, the frequent ECG preprocessing steps (filtering, segmentation, outlier removal), the recurrent DL models proposed in literature for specific tasks and also bibliometric information. The review presented in chapter 3 was continuously updated throughout my PhD with relevant publications until June 2019. To the best of my knowledge, a review on this topic has not been performed before, the extracted information from the 106 studies, reviewed in full, revealed a high heterogeneity across the structure of the DL models, therefore, the need for standardising certain hyperparameters has been
identified. Moreover, the proposed DL architectures in chapters 4 and 5 were chosen in accordance with the findings reported in the review.

Previously published reviews focused on either specific applications such as DL for ECG-based biometrics or arrhythmia detection [27], [28]. Other reviews focused on DL applied in cardiology [29], considering multiple types of input data: structured data, signals and images, however with little emphasis on the analysis of ECG signals. Another review on computational techniques for ECG analysis and their contribution to medical devices was presented by Lyon et al. in [30], [31], however, the emphasis in this review was on the traditional ML models and computer simulations for ECG analysis, very little was presented about DL technologies.

In this research, a quantitative analysis (i.e. counting, measuring, comparing quantities, computing simple statistics) was performed on the data extracted from the reviewed articles. In order to select and highlight significant and promising areas of research, a bibliometric analysis was performed. Bibliometrics is the application of quantitative analysis and statistics to publications such as journal articles and their accompanying citation counts. Therefore, bibliometrics complement qualitative indicators of research impact. Although many types of bibliometric measures can be derived, in this research three types of bibliometric networks have been explored (i.e. bibliographic coupling with respect to the referenced documents, bibliographic coupling with respect to the referenced sources and keywords’ co-occurrence) using the VOSviewer visualisation tool [32]. The bibliometric analysis revealed the diversity of ECG applications and the most recurrent DL architectures, but also impactful publications and preferred publication sources in this field of research.

Moreover, different visualisation methods were employed to synthesise the heterogeneity and to investigate the connections between different hyperparameters characterising the proposed DL models in the reviewed studies.

In study 2, a cohort of 8 (i.e. 27-58 years old) individuals underwent continuous glucose and ECG monitoring carried out in free-living conditions using wearable devices in order to investigate the feasibility of developing a personalised, non-invasive, nocturnal low glucose detection system. In study 3, a cohort of 25 (over 60s) healthy individuals were continuously monitored during a 24 or 36 hour visit at University Hospitals Coventry & Warwickshire, in a calorimeter room, having recorded continuous glucose levels and ECG signals during their hospital visit, using the same wearable devices as for study 2. Similar DL models have been employed for both studies, however, unsupervised learning techniques have been employed in study 3, for pretraining the proposed DL models, to compensate for the scarcity of the available data.
Recent clinical studies investigated the associations between certain ECG parameters and the glucose levels in both healthy and diabetic subjects [33]–[38], and showed that blood glucose concentration could affect the electrical activity of the heart. Based on these findings, the studies 2 and 3, presented in chapter 4, leveraged the representational power of DL methods to automatically detect patterns in the input ECG that are associated with a low glucose level. Previous attempts that aimed to develop ECG-based systems for low glucose detection, were trained on cohorts of participants, failing to capture the inter-subject variability in the ECG and the individual’s ECG alterations induced by low glucose events. Therefore, studies 2 and 3 introduced a novel method, following a personalised medicine approach, for low glucose detection to tackle the inter-subject ECG variability. Studies 2 and 3 also aimed to advance the understanding of the individual ECG changes induced by low glucose levels by proposing different visualisation methods for explaining how the DL models reached certain conclusions.

In study 4 (presented in chapter 5), a public dataset comprising of long-term ambulatory ECG recordings corresponding to individuals diagnosed with severe CHF and healthy individuals (33 subjects in total) was used to investigate the performance of an automatic CHF diagnostics system using DL methods. In study 4, an inter-subject evaluation strategy was employed, in which ECG excerpts corresponding to one subject were considered either for training or testing the proposed model. This evaluation method simulates real-life conditions, in which ECG recordings of a new patient might not be available in advance, to be considered for the model development. Moreover, chapter 5 presents the advantages of DL techniques for CHF detection over the previously proposed models based on traditional feature-based ML methods [8].

Diagnosis of CHF using DL methods was chosen as a case study not only because it is a serious condition with high prevalence in developed countries, but it also represents an illustrative example of a highly researched problem in the machine learning community. The focus during recent years was on finding the most informative features for discriminating between healthy persons and CHF patients, an application that can benefit from the representation learning capability of the DL methods.

In study 5 (presented in chapter 6), a public dataset comprising of 500 subjects with high quality ECG recordings [4]–[7] (≈ 7 hours of ECG/subject) was used to determine whether certain HRV features computed on a 5-minute ECG excerpt can be inferred from a short-term, (≈ 1 min ECG) raw ECG recording.

Recently, HRV analysis has become a widely-used, non-invasive method for the
assessments of the autonomous nervous system in diverse fields of research. Many studies showed that HRV reflects the self-regulation capacity (i.e. behavioural, cognitive and emotional processes), therefore HRV has been employed as a diagnostic or predictive bio-marker for diverse clinical outcomes, such as death in chronic congestive heart failure [39], myocardial infarction, mental stress detection [40], risk of accidental falls in hypertensive patients [41], an early warning sign of diabetic nephropathy [42], depression [43] and many others.

In contrast, the focus of study 5 is to investigate whether several HRV features computed on a 5-minutes ECG excerpt could be inferred using a shorter ECG interval. Therefore, the computation of specific HRV features might be bypassed, and real-time ECG applications could be facilitated due to shorter ECG segments used for the analysis [44].

1.5 Structure of the thesis

Chapter 1: presents the identified research gaps, the research questions, aims and objectives of this thesis. Furthermore, it presents an overview of the investigated case studies and employed methods. Finally, it presents an outline of the thesis.

Chapter 2: presents a brief review of the current data mining-based methods used for time series analysis (as classification and regression, feature extraction) together with some of the limitations of these techniques. Furthermore this chapter provides a motivation for the recent utilisation of DL for Time Series Classification (TSC) tasks. Furthermore, this chapter introduces state of the art DL modules and techniques that are also employed in the case studies presented in this thesis. Finally, it presents the key aspects and challenges that might occur when training DL models.

Chapter 3: provides an introduction to ECG signal physiology and anatomy, the standard devices used for the ECG acquisition, and certain aspects related to the sources of ECG signal variability. Furthermore, this chapter provides a comprehensive literature review on DL methods applied to ECG recordings published in recent studies. The literature review presents the recurrent applications, methods and the specific architectures that were used for ECG analysis in the reviewed studies. Furthermore, this chapter presents a bibliographic analysis of the recent publications in this field, highlighting the recurrent keywords’ co-occurrence, the preferred publication sources and the most cited publications.
Chapter 4: presents two experimental case studies presenting a personalised solution for low glucose detection using ECG recordings. The first study was carried out in free-living conditions whereas the second case study was carried out in a controlled environment, however over a shorter period. This chapter introduces novel methods for the automatic, non-invasive, non-intrusive low glucose detection in healthy individuals, using wearable sensors, based on ECG analysis. Moreover, the chapter presents an unsupervised learning method for pretraining the DL models that facilitates data visualisation in a lower dimensional space and might boost the performance of the low glucose detection system.

Chapter 5: presents a similar DL-based method as presented in chapter 4, however, employed for a different case study: the diagnosis of CHF, carried out on a benchmark dataset.

Chapter 6: presents a secondary analysis of a public dataset of ECG recordings and computed HRV features. This chapter presents a time series regression framework, for the estimation of several HRV features from raw ECG signal.

Chapter 7: summarises the main conclusions presented in this thesis, and provides some future work directions based on the identified limitations and future opportunities.
Chapter 2

Methodology background

2.1 Chapter overview

This chapter provides a methodological background that is relevant to the thesis. It comprises a brief review of the common methods used for time series analysis (such as classification and regression, feature extraction) together with some of the limitations of these techniques. Further, having established the state-of-the-art data mining-based methods, the chapter discusses the success of DL methods for various tasks in diverse domains and motivates their recent utilisation in TSC tasks. Section 2.4 introduces the state-of-the-art DL modules and techniques, that are also applied in this thesis. Further, section 2.6 discusses the most important aspects and possible issues when training and building DL models.

2.2 Background on time series and sequences

Time series are ubiquitous with applications in many domains: video processing, finance, genomics, health informatics, and many other. Before proceeding to the presentation of the common tasks when dealing with time series, a formal definition of time series and sequences, in general, is provided, that is used in the scope of this thesis.

In general terms a time series is usually defined as a group of sequential samples:

\[(x^{(1)}, x^{(2)}, ..., x^{(N)})\]

where \(N\) is the number of collected time series, and each \(x^{(i)} \in \mathbb{R}^{T_j}\), \(T_j\) is the number of samples in each series. Sometimes, \(T_j\) might be the same for each time series \(x^{(i)}\) and the sampling rate is uniform, i.e., the interval between two consecutive samples
from a time series $x^{(i)}$ is the same across the whole series.

Sequences, are a collection of tuples:

$$\{ \{ x^{(i)}_j, y^{(i)}_j \} \}_{j=1}^{T_i} \}_{i=1}^{N_i}$$

Here $T_i$ denotes the number of observations of the $i^{th}$ sequence, which is different among the sequence collection, $y^{(i)}_j$ denotes the timestamp of the $j^{th}$ observation of the $i^{th}$ sequence and the $x^{(i)}_j$ are the variables associated with that timestamp, thus $x^{(i)}_j \in \mathbb{R}^P$. The main difference between conventional time series and sequences is that the interval between observations is informative and the frequency of observations represents important features [45] [46].

Further, in this thesis, the sequences used in the discussed case studies represent equally spaced samples from a continuous real-world process; thus, they are referred to as time series. Moreover, the input time series is associated a target time series, denoted as $y^{(i)}$, thus $(x^{(1)}, x^{(2)}, ..., x^{(N)}) - (y^{(1)}, y^{(2)}, ..., y^{(N)})$. Therefore, instead of having tuples of observations and timestamps the input data throughout the thesis represent a set of examples where each example is a tuple of (input time series, target time series). However, in all the examples discussed in the thesis, the target time series, $y^{(i)}$ is a single data point. The TSC task is therefore defined as the problem of identifying the target value $y^{(i)}$ given the input time series $x^{(i)}$. In some of the case studies the input time series is accompanied by other structured covariates, thus the complete representation of the considered input examples can be defined as: $(x^{(1)}, \text{var}_1, \text{var}_2, ..., \text{var}_m), (x^{(2)}, \text{var}_1, \text{var}_2, ..., \text{var}_m), ..., (x^{(N)}, \text{var}_1, \text{var}_2, ..., \text{var}_m)$ each input example associated with a single point target $(y^{(1)}, y^{(2)}, ..., y^{(N)})$. Moreover, the length of each time series $x^{(i)}$ is fixed and equal for all the input examples. Also, throughout the thesis, the time series are real-valued and have no missing values. The time series analysis was performed offline, and the assumption was that the target outputs (i.e. $y^{(i)}$) were independent.

While the methods described in this thesis are applied to sequences that have an implicit temporal aspect, they are also applicable to non-temporal data such as language or genomics.

### 2.3 Background on TSC

Despite the variations in the format of sequential data, in general, time series analysis tasks fall into two categories: learning representations and making predictions [45]. Time series are mostly high-dimensional data, thus applying algorithms to such
raw data is very expensive and often gives unsatisfactory results due to the curse of dimensionality [47]. Therefore, developing representation learning techniques that can capture relevant, high-level information and reduce the dimensionality of the data can overcome this problem. Moreover, capturing the temporal component of the time series makes them even more challenging to analyse and model. Finding the right representations is an important research area, as the performance of the algorithms that extract patterns from the raw data depends heavily on the representation of data they are given. The predilect tasks for time series data that leverage different representation learning methods include clustering, classification, motif discovery, query by content and anomaly detection. Recent applications of time series learning in natural language translation, image captioning and handwriting recognition led to the development of sequence to sequence models that try to predict the next elements of a sequence from a predefined set of elements. Another active research area is time series forecasting, where the goal is to predict the future values of a time series based on the data collected up to the present. A comprehensive review on time series forecasting is presented in [48].

Out of the possible tasks when dealing with time series data, the sequence classification problem has been extensively studied, and a great number of algorithms have been developed. A brief survey of these algorithms is presented further, in section 2.3.

However, the problem of TSC has been recently explored in the deep learning community with a few applications in the biomedical domain. Thus, this thesis focuses on building DL models for TSC, with applications to ECG signals.

2.3.1 Traditional, data mining approaches for TSC

The problem of sequence classification has been thoroughly explored by the data mining community, providing detailed systematic reviews on methods and comprehensive unified validation methods [47], [49]–[54]. An extensive repository of time series data (UCR) was made publicly available by Prof. Keogh from University of California - Riverside and it contains 128 TSC datasets and 30 multivariate time series datasets [55]. Moreover, to facilitate a systematic performance comparison of the proposed algorithms, for each UCR archive dataset, the list of algorithms used for classification, their implementation and the corresponding classification performance is publicly available at [55]. Therefore, hundreds of algorithms have been proposed in the past 5 years to solve the TSC task.

The most popular methods for TSC in the field of data mining concentrated on determining novel distance measures that could be evaluated with different classi-
The benchmark distance measures are the Euclidean distance and Dynamic Time Warping (DTW) [50]. Several other distance measures have been proposed such as variants of dynamic time warping, edit distance-based measures, longest common subsequence and move-split-merge; however, Lines and Bagnall showed that there is no single measure that significantly outperforms DTW [56]. Moreover, they showed that ensembles of 1-NN classifiers (using different distance measures) significantly outperformed the individual classifiers, therefore recent contributions have been focused on developing ensemble methods [51], [53], [57], [58].

Recently, proposed ensembles comprised of different types of classifiers (decision trees, Support Vector Machine (SVM)s, neural networks) and they shared a data transformation phase into alternative representations where discriminatory features could be more easily detected. Specifically, the time series are transformed into different feature spaces using methods such as: the shapelet transform [51], DTW features [58], features extracted from symbolic representations [53] and more recently, proposed by Bagnall et al. [57] (COTE), a combination of time, frequency, change and shapelet transformation features. Lines at al. [59] extended COTE to HIVE-COTE which comprised of a hierarchical voting system, that significantly overperformed COTE. However, HIVE-COTE (an ensemble of 37 classifiers) required huge computational resources; thus it does not represent a practical solution for large, real-life datasets. Moreover, the final decision of the HIVE-COTE classifier represents another great disadvantage of this method, as the final output was obtained as a vote of the 37 different classifiers (one of the 37 classifiers has a time complexity of $O(n^{2} * l^4)$, where $n$ - number of time series in the dataset and $l$ represents the length of the time series).

### 2.3.2 DL for sequence representation learning

Having established the current state-of-the-art methods for TSC based on data mining techniques, this section provides a motivation for the utilisation of DL based methods for TSC tasks. The data mining based approaches described above required heavy crafting on the data preprocessing, feature extraction and selection, which require high computational costs. However, handling high dimensional data and complex inputs such as images and videos are predilect tasks in the computer vision field. Approaches that attempt to create lower representations of the complex inputs manually do not represent a feasible method for such high dimensional data. Problems such as object recognition in images and activity classification in videos have been successfully tackled in recent research by employing DL methods [22],
Specifically, Convolutional Neural Networks (CNNs) have revolutionised the field of computer vision as initially demonstrated by a DL based model proposed by Krizhevsky [22] in ILSVRC-2012 competition, that managed to achieve more than 10% reduction in classification error in comparison to the second-best entry, a non-DL based method.

Following the success of DL methods in the computer vision field, DL came to the attention for natural language processing and speech recognition problems. Recently, DL methods achieved impressive results for tasks such as: machine translation [62], speech recognition [63], [64], document classification [65], word embeddings [66], handwriting synthesis and recognition [67]. It’s noted that natural language processing and speech recognition tasks share a sequential type of input (i.e. a sentence can be seen as a sequence of words, and similarly speech can be decomposed into individual, consecutive frames). Therefore, the achievements obtained in the applications mentioned above and the similarity of the high dimensional inputs drove the attention of the research community towards the applicability of DL methods for TSC problems. Moreover, recent developments in graphical processing units (GPUs) enabled the efficient training of neural networks containing thousands or even millions of parameters. Nowadays, not only can neural networks (i.e. DL modules) be trained efficiently, but they are able to automatically learn hidden discriminative features from raw data in an end-to-end manner.

Figure 2.1 presents a view on the feature design effort versus the specificity of these features to the dataset when handling time series data. Designing a good set of discriminative features for different applications and datasets often requires a significant amount of time. General transformations such as Principal Component Analysis (PCA) or DTW are computationally cheap, but their limitation is that they do not account for the domain, mode of collection or context of the dataset. The features designed by data analysts are more specific to the dataset; however, a good set of features is obtained iteratively, and usually, these features need to be redesigned for new datasets. Finally, the features designed by the domain experts are the most descriptive, but are the most expensive and require the highest effort. In contrast to these approaches, DL methods can obtain specialised features without the need to involve the domain experts in the design as they have a feedback loop to adjust the representations. The domain knowledge is compensated by a complex hypothesis space (with millions of parameters) that can be searched over with a relative efficiency [68].

Several recent studies presented systematic reviews on the performance comparison of DL methods and the state-of-the-art data mining-based methods [69], [70].
Fawaz et al. [69] established that a significant statistical difference exists in the performance of the classifiers (DL vs traditional data mining-based). Moreover, they showed that not only DL methods applied for TSC were able to outperform the NN-DTW methods significantly, but they also showed that the performance results obtained with DL methods were not significantly different to those for very complex algorithms such as COTE or HIVE-COTE. Although, surprisingly, the TSC classification community seems to neglect these results so far, chapter 3 presents a review of the DL methods applied for ECG analysis. The results of this review further demonstrate the feasibility of applying DL to TSC, and in particular to ECG applications.

Section 2.4 provides some background on the main types of DL architectures that have been proposed for TSC tasks, that have been employed for ECG analysis throughout the thesis and that have been employed in the reviewed studies in Chapter 3. Moreover, section 2.4 introduces a method for inspecting the sections in the provided input that are important for obtaining class predictions, therefore providing transparency on how the networks are able to learn the mapping function.

### 2.4 Modules of DL used for time series data

Before introducing different types of DL modules, a definition for DL methods is attempted. The previous section advanced the idea that certain ML tasks can be
solved by designing the right set of features. However, sometimes it is difficult to compose a good set of features (i.e. a good representation), due to the complexity of the underlying data; usually, the manual design of the features requires a great deal of human time and effort. As outlined by Goodfellow et al. in [15], one solution to this problem is to use ML to discover not only the mapping from the given representation (features) to the output but also the representation itself.

DL solves the representation learning problem by learning hierarchical representations of the input data. Specifically, DL allows a computer to build complex concepts out of simpler ones [15]. For example, in order to recognise objects in images, DL algorithms would first learn simple concepts such as edges to define contours that in turn would define complex objects.

In general, a deep neural network is composed of $L$ parametric functions, referred to as layers, where each layer is considered a different representation of the input space [69]. Each layer contains a set of 'neurons' that can be seen as computational cells, that receive an input and produce an output that is further passed through a non-linearity. The non-linearity is required to extend linear models to represent non-linear functions. A few examples of the commonly used non-linear transformations include: sigmoid, tanh and ReLU. Different DL modules or architectures are defined by the way in which neurons are organised in each layer, and by how the output is passed to the next layer. Many DL architectures were proposed in the literature in the recent years; therefore, the algorithms presented below do not represent an exhaustive list. Section 2.4 aims at introducing the most recurrent DL algorithms that have been employed for TSC tasks. The modules presented below represent widely adopted DL models for end-to-end learning, employed with predilection in biomedical applications.

### 2.4.1 Multilayer Perceptron (MLP)

The MLP represents the simplest, classical DL module. DL networks without cycles are also called Feedforward Neural Network (FNN)s. The MLP is the most widely used form of FNN [71], [72]. An example of an MLP is shown in Figure 2.2, it comprises of units (neurons) are arranged in layers, with connections feeding forward from one layer to the next one. The neurons in layer $l_i$ are connected to every neuron in layer $l_{i-1}$ with $i \in [1, L]$. The connections between the neurons are modelled by weights and usually the summed activation arriving at a neuron is passed through a non-linearity, called an activation function. An MLP can be considered as a parameterised function that maps an input vector to an output vector, it has been demonstrated that with a sufficient number of neurons it can
approximate any continuous function [71].

2.4.1.1 Forward pass

The activation function for a hidden unit \( h \) that has \( K \) inputs denoted as \( a_h \) is given by:

\[
a_h = \sigma_h\left(\sum_{i=1}^{K} (w_{ih}x_i + b)\right).
\]  

(2.1)

The weight from unit \( i \) to \( j \) is denoted as \( w_{ij} \) and \( b \) is a bias term. Two common activation functions \( \sigma \) are either the sigmoid

\[
\sigma(x) = \frac{1}{1 + e^{-x}}
\]  

(2.2)

or the Rectified Linear Unit (ReLU) [22]

\[
\sigma(x) = \max(0, x).
\]  

(2.3)

Usually, \( b \) is either 0 or a small real number, such as \( 10^{-2} \) to \( 10^{-5} \).

2.4.1.2 Output layer

The output layer \( y \) is given by different functions depending on the target values (classification/regression). For regression tasks, usually, the output is given by the activation units in the output layer, identical to the activations in the hidden units. For classification tasks, the output layer receives the activations from the previous layer and gives a probability distribution over the class variables in the dataset. The common function used to normalise the output activations to \((0,1)\) interval, so that the obtained values can be interpreted as probabilities is the softmax function [73] as in equation 2.4.

\[
p(C_k|x) = \hat{y}_k = \frac{e^{a_k}}{\sum_{i=1}^{K} e^{a_i}}
\]  

(2.4)

where \( K \) is the number of classes and \( \hat{y}_k \) is the estimated class probability for the input \( x \) for class \( k \). Therefore the forward pass feeds the input through the network, activates the network and finally, using a softmax function it produces a score for each class.

2.4.1.3 Objective function

The parameters of the MLP need to be adjusted so that the predicted score matches the target. Therefore a cost or objective function needs to be defined that is min-
imised using an optimisation algorithm. The most widely used objective function for multi-class classification problems is the cross-entropy error \[71\], as defined in equation \[2.5\]:

$$L(\Omega) = \sum_{x,y \in S} \sum_{i=1}^{K} y_i \log(\hat{y}_i)$$  \hspace{1cm} (2.5)$$

where \(S\) defines the set of (input, target) examples in an optimisation step. For regression tasks the objective function is usually defined as the Mean Square Error (MSE) shown in equation \[2.6]\:

$$L(\Omega) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$  \hspace{1cm} (2.6)$$

where \(n\) represents the number of input and target examples used in an optimisation step and \(\Omega\) the parameter space, thus \(w \in \Omega\).

### 2.4.1.4 Backward pass

The loss functions are minimised to learn the weights \(w\), defined in equation \[2.1\], using a gradient descent method:

$$w = w - \alpha \frac{\partial L}{\partial w}$$  \hspace{1cm} (2.7)$$
where $w \in \Omega$ and $\alpha$ is a hyperparameter defining the learning rate. Therefore the parameters $w$ are adjusted such that the loss function $L$ is minimised. To compute the partial derivatives, the backpropagation algorithm is employed \cite{74}. Briefly, the algorithm employs the chain rule to compute the partial derivatives $\frac{\partial L}{\partial X}$ and $\frac{\partial L}{\partial W}$. Let $Y_n = W_n X_{n-1}$ and $X_n = \sigma(Y_n)$, where $W_n$ is a weight matrix whose number of columns is equal to the dimension of $X_{n-1}$ and the number of rows is equal to the dimension of $X_n$, $\sigma$ denotes the activation function, and $Y_n$ is the vector of weighted sums (i.e. total inputs to layer $n$). Applying the chain rule to these equations the backpropagation equations are:

\[
\frac{\partial L}{\partial Y_n} = \sigma'(Y_n) \frac{\partial L}{\partial X_n} \tag{2.8}
\]

\[
\frac{\partial L}{\partial W_n} = X_{n-1} \frac{\partial L}{\partial Y_n} \tag{2.9}
\]

\[
\frac{\partial L}{\partial X_{n-1}} = W_n^T \frac{\partial L}{\partial Y_n} \tag{2.10}
\]

The simplest minimisation algorithm is the gradient descent algorithm that updates $W$ iteratively as in equation 2.7. Commonly, equation 2.7, requires a randomly sampled mini-batch (between 20 and 300 examples) of data to compute the average gradient.

### 2.4.2 CNN

CNNs were firstly introduced by LeCun \cite{75}; they are a type of neural networks that are specialised in the processing of grid-like data \cite{15} such as images or time series. CNNs have been very successful in many practical applications such as image recognition, image segmentation, natural language processing, speech recognition, handwriting generation and many others, as presented in section 2.3.

CNNs have three fundamental properties, inspired by how the visual cortex works: local connectivity, invariance to location and shared weights \cite{75}. The architecture of a typical CNN comprises of convolutional layers, non-linear layers and pooling layers, as shown in Figure 2.3. The neurons in a convolutional layer are organised in a feature map; each neuron is connected to a local ‘patch’ in the feature maps of the previous layer through a set of weights called filters or kernels \cite{60}. The result of this local sum is passed through a non-linearity, most commonly a ReLU function. The role of the pooling layers is to merge similar features into one. At each pooling layer, maximum or average subsampling of a local patch in a feature
map is performed. The same backpropagation algorithm applied to train the simple neural networks is applied to train the weights in a CNN.

The convolutional layer consists of a set of learnable filters (i.e. weights). Usually, filters are small in terms of width and height, but they extend through the whole depth of the input volume, as shown in Figure 2.3 [76] (illustrated by the small, blue sections in the feature map space - the green parallelepiped). During the forward pass, the filter 'convolves' with the input; in other words, the filter slides across the input and dot products are computed between the filter values and the input at any position. Each filter produces one feature map; therefore, the output of a convolutional layer represents an output volume, given by the stacked feature maps. A comprehensive guide regarding the convolution arithmetic for DL is presented by Dumoulin and Vision in [77]. Briefly, three parameters determine the feature map size: the kernel size, the zero padding and the stride. Sometimes it is convenient to pad the input with zeros, which allows controlling the spatial size of the output volume. The stride specifies the number of values to slide the filter by. Depending on these three hyperparameters, the output feature map size is given by: \((W - F + 2P)/S + 1\), where \(W\) - the size of the input, \(F\) - the size of the filter, \(P\) - the padding size, \(S\) - the stride size. Figure 2.4 presents a simplified 1-D convolution operation, performed in a convolutional layer between a 1-D input and a 1-D filter when considering no pooling and different stride values. The edges between input and output neurons represent the receptive field of the output neurons, which is 3 in this example and is given by the filter size.

Similarly, the output size of a pooling operation is given by the stride, input size and pooling size; it is computed as \((W - F)/S + 1\).

The input dimension, which is 1-D instead of 2-D, and the filters which would
2.4.2.1 Visualising what is going on inside a CNN

Neural Networks are often criticised for not being transparent on how they reach certain conclusions. In particular, deep networks are considered hard to interpret, an issue that can be even more pronounced in biomedical applications, as often clinicians request some interpretable evidence on how these techniques work.

Visualising what CNN networks learn and providing insights on how the CNNs make the predictions are topics of great interest in the research community.

Two of the most common methods for inspecting what the CNN is learning are listed below:

- visualising the layers activations

The easiest and most straight-forward visualisation method is to plot the ac-
tivations of the CNN during the forward pass. This method also allows for visualising potential 'dead' filters that result in activation maps that contain mostly zeros. This situation can especially occur when employing the ReLU activation function. [76]

• visualising the first convolutional layer filters

Plotting the learned weights might also be useful, especially the ones in the first convolutional layer. These filters are the easiest to interpret, as they were convolved directly with the raw input. Noisy patterns in the learned filters might be an indicator for undertrained networks or overfitting [76].

2.4.2.2 Visualising which input subsequences are class discriminative

This section presents the use of Class Activation Map (CAM) proposed by Zhou et al. [78] and the improved version called Gradient Class Activation Map (Grad-CAM) introduced by Selvaraju et al. [79] for localisation and visualisation of the discriminative sections in the input of a CNN (can be either 1-D or 2-D). The CAM method [78], implies slightly modifying the original CNN architecture by replacing the fully connected layer/layers with a Global Average Pooling (GAP) layer. However, the downside of this method is that the complexity and performance of the CNN is traded for more interpretability and transparency. Wang et al. [70], employed for the first time CAM method for TSC, presenting the individual input time series as a heatmap, where more intense colours were more class discriminative.

Briefly, the CAM method indicates the discriminative regions in the input that were used by the CNN when making the predictions. Let $A_k(t)$ be the $k^{th}$ activation map in the last convolutional layer, at spatial location $t$. Then, the GAP neuron for activation $k$ is computed as: $GAP_k = \sum_t A_k(t)$. Therefore, for a given class $c$ the input to the softmax $S_c$ is $S_c^k = \sum_k w_c^k GAP_k$, where $w_c^k$ is the weight corresponding to class $c$ for the activation map $k$ [78]. Basically, $w_c^k$ indicates the importance of $GAP_k$ for class $c$. The final class score is obtained as a weighted average of the activations in the last layer as:

$$S_c = \sum_t \sum_k w_c^k A_k(t). \quad (2.11)$$

Then the CAM for class $c$ can be defined as:

$$CAM_c(t) = \sum_k w_c^k A_k(t). \quad (2.12)$$
By upsampling the CAM to the input size and by overlapping the CAM over the input, regions that are the most relevant for the classification can be identified.

Grad-CAM as proposed by Selvaraju et al. [79] enhanced the CAM method, by combining the last CNN activation maps with the gradient signal, such that it does not require any alterations of the original CNN. The Grad-CAM represents a strict generalisation of CAM method, a detailed demonstration is provided in [79], therefore Grad-CAM is applicable to a broader range of CNNs, including those containing fully-connected layers. In Grad-CAM the weights used in the linear combination of the forward activation maps are computed as the global average pooling of the gradients of the score for class $c$ with respect to the last activation maps. Specifically, the method implies the computation of the gradient of the score for class $c$, $y^c$, with respect to the feature maps $A$, that are global-average-pooled to obtain the weights $\alpha^c_k$. Therefore, the weights $\alpha^c_k$ are computed as follows:

$$ \alpha^c_k = \frac{1}{m} \sum_{t=1}^{m} \frac{\partial y^c}{\partial A_k(t)} $$

(2.13)

where the $\alpha^c_k$ capture the importance of the activation map $k$ for the target class $c$. The Grad-CAM heatmap for a class $c$, is obtained as a weighted combination of feature maps, followed by ReLU, shown in equation 2.14:

$$ \text{Grad-CAM\_map}_c = \text{ReLU} \left( \sum_k \alpha^c_k A_k \right). $$

(2.14)

Throughout the thesis the Grad-CAM method was employed to derive the class activation map for class $c$, which indicates the importance of the activation at a temporal location $x_i$; that lead to the classification of the input time series as a class $c$.

### 2.4.3 Recurrent Neural Network (RNN)

RNNs are another type of DL model; therefore, they are organised in layers of neurons, but their specificity is that they can capture the dynamics of the input sequences through cycles between the nodes. Unlike regular neural networks, RNNs have an internal state that can retain information from arbitrarily past context.

The main limitation of FNNs is that they rely on the assumption of independence among the examples in the training and testing datasets [80]. If the data points are related in time or space, then past information is lost when employing an FNN. Unlike FNN, RNNs can pass information across sequence steps, therefore they can...
model input/output consisting of sequences of elements that are not independent [80], [81]. Another advantage of RNNs is that they can capture long-term dependencies, overcoming the main limitation of Markov based models, (i.e. assumes a discrete number of states and that future states depend only on the current state, not on the events that occurred before). Moreover, similar to CNNs, given a fixed set of nodes, edges, activation function, the RNNs are differentiable end-to-end; therefore, they can be optimised with respect to a loss function in a relative amount of time. Also, RNNs are not limited to time-based sequences; they have been employed successfully for non-temporal data, such as genetic data.

A simple example of a RNN is shown in Figure 2.5. At time $t$, the node receives as input the current input point $x_t$ and the data from the previous hidden node state $h_{t-1}$. The output $y_t$ at each time $t$ is calculated given the hidden node values $h_t$. Therefore, the input $x_0$ can influence the current time step, but also on the future time steps due to the recurrent connections [80]. The equations that specify the calculations at each time step in the forward pass for a network similar to the one presented in Figure 2.5 are given by equation 2.15 and 2.16:

$$h_t = \sigma(W_{xh}x_t + W_{hh}h_{t-1} + b_h) \quad (2.15)$$

$$y_t = \text{softmax}(W_{hy}h_t + b_y) \quad (2.16)$$

where $W_{xh}$ is the matrix of weights between the input and the hidden layer, $W_{hh}$ is the matrix of the recurrent weights between the hidden layer at consecutive time steps, $W_{hy}$ is the matrix of the weights between the hidden layer and the output layer and $b_h$ and $b_y$ are two bias vectors.

Training and learning with RNNs has been considered to be very challenging due
to the difficulty of long-term dependencies, as explained by Hochreiter et al. in [83]. The two problems that might occur are the vanishing or exploding gradients, due to backpropagating the errors across many time steps. The problem is caused by the weights that are shared across the time steps, therefore the contribution of the input at time $t$ to the output will either explode (tend to inf) or vanish (tend to 0), thus the derivative of the error with respect to the input will either explode or vanish [80], [81]. To account for these problems several methods have been proposed in the literature: gradient clipping and regularisation [84], truncate the backpropagation through time [84], [85].

The most successful and most widely used type of RNN architecture was first published by Hochreiter and Schmidhuber in 1997 [83]; the breakthrough that managed to improve the gradient flow in the RNN was the introduction of the memory cell. The cell, called Long Short-Term Memory (LSTM), acts as a computational unit that replaces the traditional hidden node in the hidden layers. The LSTM architecture consists of a list of recurrently connected subnets or memory blocks. Each memory block can contain one or more memory cells. Figure 2.6 presents the structure of a single LSTM memory cell which is also the chosen RNN structure used in this thesis due to its ability to handle long-term dependencies while bypassing the exploding and vanishing gradients problem. As shown in Figure 2.6, a memory cell is built from simple nodes and multiplicative gates that allow the LSTM to store and access information over long periods of time. Therefore, a traditional LSTM cell is comprised of the following nodes and gates and performs specific calculations at each time step [82]:

Figure 2.6: LSTM memory cell, based on [82].
• Forget gate layer
  This layer decides what information will be discarded from the cell state \((C_t)\) in Figure 2.6. The decision is made by a sigmoid layer, called a 'forget gate layer', given in equation 2.17.

\[
f_t = \sigma(W_f[h_{t-1}, x_t] + b_f),\tag{2.17}
\]

where \(W_f\) is a weights matrix corresponding to the forget gate layer and \(b_f\) a bias matrix. The layer considers the previous cell output \((h_{t-1})\) and the current cell input \((x_t)\) and outputs a value between 0 and 1, where 0 means that no information is preserved in the cell state and 1 means that the entire cell information is preserved.

• Input gate layer
  The next step is to decide what information will be updated in the cell state. This is achieved through the sigmoid layer \((i_t)\), given by equation 2.18,

\[
i_t = \sigma(W_i[h_{t-1}, x_t] + b_i).	ag{2.18}
\]

• Cell state, new candidate values
  A tanh layer creates the vector of potential candidate values for the \(C_t\), denoted \(C'_t\), given by the equation 2.19,

\[
i_t = \tanh(W_c[h_{t-1}, x_t] + b_c).	ag{2.19}
\]

• Cell update
  The final computations for updating the cell state with the new values is performed by multiplying the old state by \(f_t\), therefore forgetting the information decided earlier, then using the input gate to filter the candidate \(C'_t\) and adding it to the remaining information in the cell, given in equation 2.20,

\[
C_t = f_t \ast C_{t-1} + i_t \ast C'_t.	ag{2.20}
\]

• Output gate layer
  This layer decides what information the cell outputs, therefore, similar to the input gate, this is achieved using a sigmoid gate,

\[
o_t = \sigma(W_o[h_{t-1}, x_t] + b_o).	ag{2.21}
\]
Figure 2.7: An RNN with a hidden layer consisting of 2 memory cells, unfolded across time, based on [80].

- **Final LSTM cell output**

  The final output is obtained as a filtered version of the cell state using a tanh gate and by multiplying it by the output of the sigmoid gate, so that only the parts decided earlier will be the final output, as shown in equation 2.22,

  \[
  h_t = o_t \ast \tanh(C_t). \tag{2.22}
  \]

Figure 2.7 presents a recurrent neural network with a hidden layer consisting of multiple memory cells (i.e. 2), shown unfolded across time. The output from each memory cell flows in the subsequent time step to the input node and all the gates (forget, input and output) of the memory cell [80], [81]. Therefore, only the cell outputs are connected to the other memory blocks in the layer, the other gate outputs are only visible within the block, as illustrated in Figure 2.7.

2.4.4 Autoencoder (AE)

Autoencoders are general dimensionality reduction methods that were initially introduced by Hinton et al. [86], who showed that high-dimensional data can be converted to a lower dimension by training a multilayer neural network with a small central hidden layer, used to reconstruct the high-dimensional input. Usually, an autoencoder involves two deterministic transformations (affine mapping and a non-linearity) and has two main components: an encoder \( f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}^{d'} \), of the type \( y = f_\theta(x) = \sigma(Wx + b) \) and a decoder \( g_{\theta'} : \mathbb{R}^{d'} \rightarrow \mathbb{R}^d \), where \( g_{\theta'}(y) = \sigma(W'y + b') = z_{\theta,\theta'}(x) \). Therefore, given a set of unlabelled inputs \( \{x_i|x_i \in \mathbb{R}^d, i = 1,2,...,n \} \), an autoencoder attempts to learn a reconstruction function such that \( z_{\theta,\theta'}(x) \approx x \).
The parameters of the reconstruction function can be found by minimising the reconstruction loss that is usually defined as a squared error, $\mathcal{L}(\hat{x}, \tilde{x}) = \|\hat{x} - \tilde{x}\|^2$ for a real-valued $\tilde{x}$. Thus, autoencoder training consists of minimising the following objective function, which is typically carried out by stochastic gradient descent:

$$\argmin_{\theta, \theta'} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(x_i, g_{\theta'}(f_{\theta}(x_i)))$$

Intuitively, a good representation $f_{\theta}(x)$ retains as much of the information that is present in the input. However, retaining as much information as possible from the input is not an adequate criterion (e.g. $y$ can be simply set to an identify function) for obtaining a useful representation, which also implies discarding the noise in the input [87], [88]. Two common strategies are usually employed for obtaining useful representations of the input: 1) enforce a compression on the representation, thus enforce $d' < d$ (under-complete autoencoders), 2) use a higher dimension for the representation than the input ($d' > d$), but enforcing a sparsity constraint (over-complete autoencoders). In this thesis, the employed autoencoder was a Convolutional Denoising Autoencoder (CDAE), that is a simple autoencoder with the following characteristics: the encoder and decoder components represent CNNs and the reconstruction function learns a mapping from a distorted version of the input to recover the non-distorted input. Throughout the thesis the CDAE was employed to counteract the scarcity of the input data for a classification task by learning a representation of the input data as a first analysis step.

2.4.4.1 Denoising Autoencoder

A denoising AE is a variant of the AE described above, which tries to reconstruct a clean output from a corrupted version of the input [88]. Recent studies showed that more robust features can be obtained by training denoising autoencoders, that were assessed for different classification tasks [87], [89]. The ‘corrupted’ version $\tilde{x}$ of the input $x$ can be obtained by applying a stochastic mapping $\tilde{x} \sim D(\tilde{x}|x)$, which is application dependent but common noise models include: Gaussian noise, Salt-and-pepper noise, Masking noise, etc.[87]. Thus, the key difference is that $z$ is a deterministic function of $\tilde{x}$ instead of $x$, as shown in Fig. 2.8.

2.4.4.2 Convolutional Autoencoder

Convolutional Autoencoder (CAE)s differ from simple AE as their weights are shared across the input for preserving spacial locality. Specifically, both the encoder and
Figure 2.8: General denoising AE diagram. The input example $x$ is corrupted to $\tilde{x}$, the AE then maps $\tilde{x}$ to $y$ and tries to reconstruct $x$.

Figure 2.9: Example of the CDAE. The CDAE is composed of convolutional and deconvolutional (convolutional transpose) layers and the latent representation($y$). The input for the CDAE is a corrupted version($\tilde{x}$) of $x$ that is used as an output. Usually, the learned parameters from training the CDAE are used to initialise a classifier for different target classes.

the decoder are given by CNNs. Therefore, the CAE architecture is similar to an AE, described in section 2.4.4 except that the weights are shared, which results in fewer parameters while taking advantage of the CNN’s capability of recognising and learning patterns in the input. The convolutional/deconvolutional layer, followed by an activation is given by [90], [91]:

$$h^k = \sigma(\sum_{l \in \mathcal{L}} x^l * W^k + b^k)$$

(2.23)

where $h^k$ represents the latent representation of the $k^{th}$ feature map, $x^l$ is the $l^{th}$ feature map of the previous layer feature maps($\mathcal{L}$), $*$ denotes a 1-D convolution, $\sigma$ is the activation function (sigmoid, ReLU, tanh) and the bias $b$ which is shared across the map. In the case studies presented further in the thesis, the performance of the AE was tested on how useful the learned representations were for a certain task of interest. Therefore, a good representation was defined as one that yields a more accurate classifier for a certain task.
2.5 Embeddings visualisation

The t-Distributed Stochastic Neighbor Embedding (t-SNE) proposed by Maaten and Hinton [92] is a technique for high-dimensional data visualisation. The authors have shown that t-SNE is capable of capturing the local structure of the high-dimensional data, but at the same it can reveal global structures in the high-dimensional data (i.e. clusters) [92]. Briefly, the t-SNE method converts pairwise distances in both data space and latent space (low-dimensional space) into probabilities that measure the similarity of two datapoints and then minimise the Kullback-Leibler (KL) divergence between those probabilities. Specifically, the pairwise distances in the data space are converted to probabilities by centring a distribution (i.e. a Gaussian distribution) over each datapoint $x_i$, obtaining the normalised density at a point $x_j$ as:

$$p_{ji} = \frac{e^{-||x_i - x_j||^2/2\sigma_i^2}}{\sum_{k \neq i} e^{-||x_i - x_k||^2/2\sigma_i^2}}$$

(2.24)

in which the variance of the distribution, $\sigma_i$, is set such that the perplexity of each $P_i$ (i.e. probability distribution over all datapoints) of each conditional distribution is equal and $p_{ij}$ is zero. Perplexity is a hyperparameter, defined by the user, usually having values between 5 and 50. The perplexity of $P_i$ is given by equation 2.25,

$$\text{Perp}(P_i) = 2^{H(P_i)}$$

(2.25)

where $H(P_i)$ is the Shannon entropy of $P_i$ [92]. Finally, the joint distribution $p_{ij}$ is given by the symmetrised conditional probabilities: $p_{ij} = \frac{p_{ij} + p_{ji}}{2n}$. This ensures that each datapoint $x_i$ makes a contribution to the cost function.

To measure the pairwise similarity of datapoints $i$ and $j$ in the low-dimensional space, a symmetric distribution is centred over each datapoint $y_i$ in the latent space. Similarly, as for the high-dimensional space the density of all other points, $j$, is measured and the result is normalised to obtain probabilities $q_{ij}$, that give the local structure of the data in the latent space. t-SNE employs a Student t-distribution with one degree of freedom for the latent space map, thus the $q_{ij}$ are given by [92]:

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq i}(1 + ||y_k - y_i||^2)^{-1}}$$

(2.26)

Having computed the probability distributions $P$ and $Q$, t-SNE minimises the KL divergence.
divergence between $P$ and $Q$, given in equation 2.27,

$$Cost = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}. \tag{2.27}$$

Therefore, the objective function focuses on modelling similar points in the original, high-dimensional space, as similar or close together in the low-dimensional space, preserving the local structure of the data.

Other methods such as PCA or AE can also be considered as dimensionality reduction techniques, however it has been shown that t-SNE outperforms both methods [93]. The better performance of t-SNE in comparison to PCA is due to the linear nature of PCA and also PCA tries to retain large pairwise distances in the latent space, whereas t-SNE focuses on retaining mainly the local structure of the data in the latent space. The aim of an AE is to maximise the variance of the data in the latent space, in order to obtain low reconstruction errors. Therefore, AEs in general, do not construct low-dimensional data representations in which the classes are clearly separated, as this would decrease the variance of the low dimensional data.

t-SNE method has been used throughout this thesis for visualising the learned embeddings for performing different classification tasks, or for inspecting the AE learned embeddings in a lower-dimensional space (i.e. 2-D and 3-D).

### 2.6 DL networks training

Section 2.4 introduced the idea that neural networks can be trained through differentiation of a suitable objective function using gradient descent techniques. However, to ensure that the network training is effective, there are multiple quantities that are usually monitored and several aspects that should be considered, these are briefly described in the following subsection.

#### 2.6.1 Optimisation algorithms

The choice of optimisation algorithms plays an essential role in the successful training of neural networks. DL algorithm optimisation is a very active area of research, many different optimisation algorithms have been proposed in the recent years; therefore, here, only the most widely used algorithm are mentioned.

The simplest optimisation algorithm is known as steepest descent or gradient descent; it takes a small, fixed-size step in the direction of the negative error gradient,
\[
\theta_{t+1} = \theta_t - \mu \Delta \theta L(\theta).
\] (2.28)

Depending on how much data are used to compute the gradient of the objective function and perform a parameter update, gradient descent algorithms have three variants: batch gradient descent (the entire training dataset is used), stochastic gradient descent (parameter update for each training example) and mini-batch gradient descent (performs an update for a mini-batch of training examples randomly sampled from the training dataset) [94]. In practice the mini-batch gradient descent methods are preferred; the mini-batch size varies with the number of training examples, typical mini-batch sizes are set to values between \(2^6\) and \(2^{10}\).

To speed up the training process even further, a large number of more complex optimisation algorithms have been developed: Momentum [95], Nesterov Momentum [96], Adagrad [97], Adadelta [98], RMSprop (slide 29, Lecture 6 of Geoff Hinton’s Coursera class), Adam [99] and some other more.

Mini-batch gradient descent, Momentum and Nesterov Momentum manipulate the learning rate globally and equally for all parameters. Adadelta, Adam, RMSprop are methods that adapt the learning rate to the parameters, performing larger updates for sparse parameters and smaller updates for frequent parameters. As Adam was employed as the optimisation method throughout the thesis after running multiple experiments with different optimisation methods the paragraph below provides more details about its implementation.

Adam stores per-parameter, a sum of the past squared gradients \(v_t\) and an exponentially decaying average of the past gradients \(m_t\) [94]:

\[
\begin{align*}
    m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\
    v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2
\end{align*}
\] (2.29)

where \(m_t\) and \(v_t\) are the estimated mean and variance of the gradients, \(\beta_1\) and \(\beta_2\) are decaying rates. Having computed \(m_t\) and \(v_t\), the Adam’s update rule for the parameters is given by:

\[
\theta_{t+1} = \theta_t - \frac{\mu}{\sqrt{v_t} + \epsilon} m_t
\] (2.30)

The authors recommended the following general default settings for the parameters: \(\mu = 0.001\), \(\beta_1 = 0.9\), \(\beta_2 = 0.999\) and \(\epsilon = 10^{-8}\).

Other groups of optimisation methods involve the computation of the Hessian matrix (second-order partial derivatives) that describes the local curvature of the
loss function, which allows one to perform a more efficient parameter update. However, in practice, the computation of the Hessian is a very costly process in both space and time. Therefore, in practice, Adam is the current, recommended default optimisation algorithm [76].

2.6.2 Generalisation

The term generalisation refers to the ability of the ML algorithms to achieve similar performance on unseen data to the performance obtained on the training dataset [71]. Many algorithms have been proposed in the literature, in this thesis; however, only three of the most common methods for improving generalisation were used: early stopping, regularisation and distorting the input with some noise.

2.6.2.1 Early stopping

Early stopping implies using a third dataset (validation or development dataset) to estimate the generalisation error and to stop the training once the performance on the validation dataset does not improve (defined by a stopping criterion), as illustrated in Figure 2.10 [81], [100]. Typically, the validation dataset is obtained by randomly sampling examples from the training dataset (20% up to 50%). Therefore, the network is trained only on the training dataset, evaluated once in a while on the validation dataset and should only be evaluated on the test dataset when the training is finished. The error during training decreases initially on the validation dataset, however after a certain point, it plateaus and begins to rise, behaviour known as overfitting. The early stopping method uses the validation dataset to anticipate the error obtained on the unseen test dataset, therefore the validation error is used as an estimate of the generalisation error.

Among many possible stopping criteria, the criterion employed in this thesis was defined as **stop when the generalisation error did not improve in $s$ consecutive optimisation steps**. The idea behind this criteria is that when the validation error increased not only once, but during $s$ consecutive steps, this might indicate the beginning of the overfitting. The performance measure used in this thesis was the **Area Under the Curve (AUC)** and $s$ varied with the number of examples included in the training dataset.

The drawback of the early stopping method is that a percentage of the training dataset cannot be used for training, as it has to be sacrificed for the validation dataset, therefore this might lead to a decrease in the performance of the ML model.
Figure 2.10: Overfitting on training data. The error on the validation dataset initially decreases, however after several training epochs it reaches a plateau and starts to increase. The dashed line indicates the ideal point on the best performance on the validation dataset, therefore, the early stopping point can be used to stop the training process.

2.6.2.2 Regularisation

Regularisation is another technique to control the capacity of the model and prevent overfitting. Regularisation reduces the allowed parameter space of the model that usually leads to better generalisation on the unseen data. The most recurrent regularisation techniques include: imposing a sparsity constraint, introducing penalty terms to the loss function and dropout [15], [71], [76]. **L2 regularisation** and **dropout** were employed as regularisation methods in the case studies presented in the thesis.

L2 regularisation (or L2 norm or Ridge regression) penalises the squared magnitude of all the parameters of the network, a penalisation term that is added to the objective function. Specifically, for every weight \( w \) in the network, the term \( \frac{1}{2} \lambda w^2 \) is added to the objective, where \( \lambda \) is a hyperparameter that controls the penalisation strength. The intuition behind the L2 norm is that it penalises peaky weights, preferring more diffuse weight vectors [76]

Dropout, proposed by Srivastava et al. [101] proved to be a simple and very effective regularisation technique. Dropout implies keeping a neuron active with some probability \( p \) during training, or setting it to zero otherwise. For each example
in a mini-batch, a randomly sampled binary mask is applied to all inputs and hidden units in the network that were assigned a dropout regularisation. The mask for each unit is sampled independently, and the probability of keeping a neuron active (i.e. $p$) is a hyperparameter.

### 2.6.2.3 Other methods for improving generalisation

Other standard methods for improving the generalisation ability of the ML algorithms include training on more data or training on distorted inputs. A simple method for obtaining more data is to generate 'fake' data that are further added to the training dataset. These methods are called data augmentation techniques. For image inputs, such augmentation methods include distortions applied to the original image such as: rotation, translation, scaling, cropping. For time series window slicing and resampling methods have been proposed as data augmentation techniques, as used in [102]. However, the process of producing highly structured corrupted inputs is usually based on prior knowledge assumptions regarding the inputs.

A simpler and more general method applicable to any type of input is to add noise into the network inputs during training (also called training with jitter) [71], [103]. Its effect is to generate noisy inputs artificially, that might affect the performance on the training dataset but might increase the performance on the validation and testing datasets. This research field is somewhat related to autoencoders and denoising, presented in section 2.4. However, in the case of denoising autoencoders, the focus is on using a local unsupervised denoising criterion to pretrain the classification network so that it learns better representations of the input space.

In practice, it has been shown that it is always better to use regularisation methods to control overfitting than reduce the number of neurons [76].

### 2.6.3 Input representation

The performance of the ML algorithms depends heavily on the representation of the input data they are given. Therefore, preprocessing of the raw input is most of the time required to get the data in a form that is both complete (meaning that the input contains the information required to predict the output) and compact [81]. However, the choice of the input representation is subjective, application dependent and there is no standard that regulates it.

In contrast, normalisation of the features of the input vector should always be performed. LeCun showed that this normalisation has no negative effect on the information contained in the training dataset and at the same time it improves the
performance of the ML models by shifting the input values into a range close to the active regions of the activation functions. The same normalisation procedure applied to the training dataset should be employed on the validation and testing datasets.

The two most common forms of input data normalisation are: mean subtraction and z-normalisation (or standardisation). Mean subtraction involves subtracting the mean across each feature in the data. Standardisation involves transforming the input features to have mean 0 and standard deviation 1 over the training dataset.

For the time series case, where the input dataset does not contain a list of examples each comprising of a set of individual features, normalisation is usually performed on each time series. In order to make meaningful comparisons between two time series, they need to be normalised, as previously demonstrated by Keogh and Kasetty in 2003 [49] and further emphasised in many other studies [52], [54]. Specifically, z-normalisation is given by equation 2.31:

\[
\begin{align*}
\mu_i &= \frac{1}{n} \sum_{j=1}^{n} x_i[j] \\
\sigma_i &= \sqrt{\frac{1}{n} \sum_{j=1}^{n} (x_i[j] - \mu_i)^2} \\
\hat{x}_i &= \frac{x_i - \mu_i}{\sigma_i}
\end{align*}
\]

(2.31)

where \(x_i\) represents a time series, \(\mu_i\) and \(\sigma_i\) are the mean and the standard deviation of the time series \(x_i\) and \(n\) is the time series length.

As shown in equation 2.31, z-normalisation scales the amplitude of the time series and removes the offset, allowing the ML algorithm to focus on the structural patterns rather then on the amplitude-driven ones.

2.6.4 Weight initialisation

All gradient descent based algorithms require the network to contain small close to 0, random initial weights for 'symmetry breaking’. The rationale for this choice is that the neurons are expected to be random and unique in the beginning so they will compute different updates and integrate as diverse parts of the network [104]. Therefore, a simple solution is to initialise the weights with values from a flat random distribution in the range [-0.1, 0.1] or a similar interval. Also, sampling from a Gaussian distribution with mean 0 and standard deviation 1 is a common
initialisation strategy, so the neurons point in a random direction. It is also possible to use small random numbers drawn from a uniform distribution, but it has been shown this has little impact on the performance of the model in practice [104].

Although effective in practice, the initialisation methods mentioned above suffer from a common problem, i.e. the distribution of the outputs from a randomly initialised neuron has a variance that grows with the number of inputs [76]. Starting from this observation Glorot and Bengio [105] proposed a normalisation method to account for the high variance of the outputs, a method that is currently known as Xavier initialisation. Specifically, the initialiser is designed to keep the scale of the gradients roughly the same in all layers, given by: $\text{Var}(w) = \frac{2}{n_{in}+n_{out}}$, where $n_{in}$ and $n_{out}$ are the number of units in the previous layer and the next layer. A more recent study [106] showed that, particularly for ReLU neurons, the variance of neurons should be $\sqrt{\frac{2}{n_{in}+n_{out}}}$.

In this thesis, the Xavier initialiser with ReLU correction was employed as an initialisation method, with only one exception where the network weights were initialised from a Gaussian with mean 0 and standard deviation 1.

Another technique developed by Ioffe and Szegedy [107], called batch normalisation provides a solution for the weights initialisation problems. Specifically, the authors proposed to explicitly normalise the network activations to have a unit Gaussian distribution. Usually, batch normalisation is performed after the fully connected or convolutional layers and before the non-linearity. Specifically, the batch normalisation applied to an activation $x$ over a mini-batch is given by the equation 2.32 and described in more detail in [107],

$$
\mu_{\beta} = \frac{1}{m} \sum_{i=1}^{m} x_i \\
\sigma_{\beta}^2 = \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_{\beta})^2 \\
\hat{x}_i = \frac{x_i - \mu_{\beta}}{\sqrt{\sigma_{\beta}^2 + \epsilon}} \\
y_i = \gamma \hat{x}_i + \beta = BN_{\gamma,\beta}(x_i)
$$

(2.32)

where $\mu_{\beta}$ is the mini-batch mean, $\sigma_{\beta}^2$ is the mini-batch variance, $y_i$ is the output of the batch normalisation of $x_i$ that is also scaled and shifted by two learnable parameters $\gamma$ and $\beta$. Throughout the thesis, batch normalisation was employed after each convolutional layers to calibrate the variance of the output neurons that grows with the number of inputs.
2.6.5 Hyperparameters

As emphasised in the previous sections, training neural networks and therefore DL models involves many design choices such as the optimisation method, architecture type and structure, connectivity type, hyperparameters. The essential hyperparameters to be set are: the initial learning rate, the learning rate decay if implemented, the regularisation type and strength and the mini-batch size. It is clear that a full grid search over the possible values and combination of hyperparameters is not feasible due to high computational costs in both time and space. Instead, several heuristic methods for the hyperparameters’ search have been proposed in the literature [46], [108], [109], such as:

- sample random hyperparameters from a predefined range of values
- a grid search over a range of predefined range of values (although Bergstra and Bengio showed that random sampling is more effective than grid search [108])
- manual tuning, especially for hyperparameters that have less effect on the performance of the ML algorithm (such as mini-batch size, momentum parameters)

Several known best practices that enable a systematic search for hyperparameters values, are listed below, practices also employed throughout the thesis [76]:

- prefer one validation split, instead of performing cross-validation
- specify the learning rate and regularisation strength values ranges on a log scale, eg: $10^{uniform(-3,1)}$
- start from the coarse values ranges, and after evaluating the performance advance to finer ranges for further tuning
- check that the best performance was not obtained for a value on the border of the specified interval
- some hyperparameters can be set according to practical recommendations [46]
2.7 Conclusions

This chapter provides a brief overview of the methods and techniques employed in this thesis. Firstly, a summary of the state-of-the-art data mining-based methods for TSC were introduced, highlighting some of the limitations of these methods and therefore providing motivation for employing DL based methods for TSC tasks. Further, the most recurrent DL modules were introduced, and section 2.6 provided detail about the training of deep neural networks process. The next chapter, 3, discusses the application of DL methods for ECG analysis, which mainly includes classification tasks, therefore ECG classification is a particular case of the general TSC task. The research questions explored in this thesis concern the application of DL methods for different ECG analysis tasks, thus all the techniques and methods presented in this chapter were applied further throughout the thesis in chapters 4, 5 and 6.
Chapter 3

Literature review on DL for ECG analysis

3.1 Chapter overview

This chapter provides an introduction about the ECG signal, briefly presenting the physiology and anatomy of the ECG, the common devices used for the ECG acquisition, and certain aspects related to the ECG signal variability sources. Further, section 3.3 provides a comprehensive literature review on DL methods applied to ECG recordings in recent studies, published in the last 3 years. The literature review presents the recurrent applications, methods, specific architectures that were used for ECG analysis in the reviewed studies. Moreover, section 3.3 presents a bibliographic analysis of the recent publications in this field, highlighting the recurrent keywords’ co-occurrence, the preferred publication sources and the most cited publications.

3.2 ECG fundamentals

3.2.1 ECG anatomy and physiology

ECG signals represent a graphic record of electrical events captured from the body surface when the cardiac muscle undergoes depolarisation and repolarisation [110]. Therefore, an ECG represents a valuable tool used in clinical practice that can aid in the diagnosis of many cardiovascular diseases.

Under normal circumstances, an electrical impulse is initiated by the sinoatrial (SA) node and spreads throughout the atria. The impulse takes approximately 50ms to travel to the next node in the conduction system, the atrioventricular node. Af-
ter reaching the atrioventricular node, there is a delay of approximately 100ms that allows the atria to complete pumping blood before the impulse is transmitted to the atrioventricular bundle. Following the delay, the impulse travels through the atrioventricular bundle and the bundle branches to the Purkinje fibres. Afterwards, the impulse spreads to the contractile fibres of the ventricle, after which the ventricular contraction begins [111]. These electrical currents can be detected and measured using electrodes placed at various sites on the body’s surface, therefore obtaining the ECG signal, as shown in Figure 3.1.

A typical ECG has three recognisable deflections or waves: the P wave, the QRS complex and the T wave, as shown in Figure 3.1. The first wave, the P wave, usually lasts about 0.08s and results from the depolarisation wave from the SA node through the atria. The QRS results from ventricular depolarisation; the average duration of the QRS complex is 0.08s. The T wave is caused by ventricular repolarisation; usually it lasts about 0.16s. Typically, the repolarisation is slower than the depolarisation, resulting in a T wave that is more spread, and has a lower amplitude than the QRS complex. The P-R interval (about 0.16s) is the time from the beginning of the atrial excitation to the beginning of the ventricular excitation. The Q-T segment (usually about 0.38s) marks the beginning of the ventricular depolarisation through ventricular repolarisation.

### 3.2.2 Acquisition

A standard ECG recording device consists of 12 leads, three electrodes from bipolar leads that measure the voltage difference between the arms or between the arm
and a leg and 9 unipolar leads. Together the 12 leads provide a detailed image of the heart’s electrical activity. Nevertheless, clinical ECG acquisition presents several advantages and disadvantages: it provides a complete, accurate picture of the heart’s activity, however at the same time it requires a large number of electrodes, and the subjects are usually asked to lie still during the acquisition. Due to these requirements, it is not very common for the ambulatory ECG to be used for real-time applications, unless the ECG was collected in the past and the studies performed are retrospective.

To address the disadvantages of ECG acquired in clinical settings, Holter systems were proposed. These systems enable long-term ECG recordings (24h or even several days) in free-living conditions, however using fewer electrodes, therefore less ECG leads. Recent body-worn sensors such as the Zephyr BioPatch [112], the Heartsense heart monitor [113] or QARDIOCORE [114] are examples of wearable ECG monitors designed to provide continuous, accurate ECG data in free-living conditions. These devices can only record 1-lead ECG signal and they either involve 2 chest electrodes or a strap that allows the monitor to be attached on the chest.

Other devices such as the KardiaMobile [115] provide a portable ECG recorder. The AliveCor Heart Monitor is a pocket-sized device containing 2 electrodes. To record the ECG with AliveCor Heart Monitor, the user needs to place 2 or more fingers from the left hand onto one of the electrodes and 1 or more fingers from the right hand onto the second electrode for at least 30s. After the AliveCor Heart Monitor has taken a reading, it is sent wirelessly by a high-frequency sound transmission to a mobile device, where it can be viewed using the AliveECG phone application. These types of devices are also referred to as off-the person ECG recording devices as they do not require the user to wear the monitors continuously, therefore they are completely non-invasive.

### 3.2.3 ECG intra- and inter-subject variability

The ECG signal presents a high degree of variability even, when recorded in normal conditions, on healthy individuals. The ECG variability can be observed both intra-subject (therefore variations in the cardiac cycles of the same subject) or inter-subject (variations between heartbeats corresponding to different individuals).

Both intra- and inter-subject variability may be caused by a series of factors such as: age, sex, body weight, chest configuration, food intake (both low and high glucose levels can produce electrocardiographic changes which can be similar to abnormal ECG), exercise, smoking and race. It has recently been shown there are differences in the ECG between different races: Caucasian, Chinese, Indian, Black.
Specifically, QRS voltages were higher in young black males compared to the others, ST amplitudes were higher in Chinese and Nigerian males than in Caucasians [116].

Moreover, the QRS axis might be affected during pregnancy, due to a commonly elevated diaphragm [117]. Furthermore, it has been found that in healthy people over 70 years of age, 30-85% of ECG are outside of the normal limits.

It is also known that many electrocardiographic and electrophysiologic parameters exhibit certain circadian patterns [118], therefore the individual circadian rhythm also contributes to the intra-subject ECG variability. It has been found that the pattern of repolarisation heterogeneity is significantly different between men and women and that both genders exhibit circadian patterns [118]. Specifically, RR intervals, T wave morphology and T wave duration are affected by the circadian rhythm, they are higher during the night, the same pattern could be observed in both men and women [118].

Standing or lying down, influences the position of the internal organs; therefore the heart’s position is also affected by posture, which will cause variations in the collected ECG signal.

Physical exercise, fatigue and emotions have a direct influence on the heart rate, which has a signature on the duration of and interval between the heartbeat deflections, mainly the distance between the QRS and T wave.

Lastly, the electrode type, number and placement can also affect an ECG recording. In the case of wearable ECG devices, chest placement of the electrodes is crucial, as different positions of the electrodes would result in different heartbeat shapes. Moreover, electrode inversion would result in the ECG waveform inversion.

Therefore, all the previously presented factors have a signature in the ECG morphology. When developing applications that consider ECG signals as input, all these factors should be taken into account, especially when formulating conclusions. Particularly for ECG-based biometric systems, intra-subject variability sources should be considered, otherwise the systems might fail to recognise an individual based on personal ECG patterns.

3.3 Review of ECG analysis applications with DL

The review presented in this section aims to synthesise the recent evidence regarding DL applications for ECG signal and to identify the optimal combination of DL methods and hyperparameters aiming to support the design of new studies and real-life applications.
3.3.1 Literature search

Potentially relevant articles on DL methods for the ECG signal were identified through a literature search in PubMed and Scopus.

Articles were searched using Boolean combinations of the following terms, denoting either different DL methods or ECG acronyms: ("deep learning" OR convolutional neural network* OR "CNN" OR "recurrent neural network*" OR lstm OR rnn) AND (ECG or electrocardiogram or EKG). Only the articles published in the past 3 years were included in this review. To minimise the likelihood of missing certain papers the following strategy has been adopted: keywords were searched in the article titles and abstract, the full keyword and its acronyms were included in the search and also truncations have been used to include grammatical variations of the same word.

3.3.2 Inclusion and exclusion criteria

Papers that were considered suitable for this review should meet all of the following criteria:

1. Original peer-reviewed journal and conference articles published between January 2017 and May 2019, written in English;

2. Studies in which the inputs of the DL models were the raw ECG signals, sometimes maybe also fused with other structured covariates;

3. Studies that clearly describe the data used for the analysis (including the number of subjects, the sampling frequency of the ECG signal, the method for training and testing the models: either intra-subject or inter-subject) that could be either from a public dataset or a private dataset;

4. Studies that clearly describe the architecture of the DL models, including the dimension of the input, the number of layers, filters and filter sizes for a CNN network or the number of layers and hidden units for a RNN, Deep Belief Network (DBN) architectures. Papers that transformed the ECG signal to other formats: other time series or images were excluded from the analysis.

3.3.3 Paper selection and data extraction

Following the search strategy described above, all the articles selected based on the keywords search were identified. Duplicated articles were excluded (i.e. articles that were indexed in both databases) and studies were shortlisted according to the
inclusion/exclusion criteria listed above by screening titles, abstracts and the full-texts.

Subsequently, relevant data were extracted from the shortlisted studies, specifically: the name of the authors, the year of publication, aim of the study (i.e. application), datasets used, main ECG pre-processing steps: filtering, normalization, heartbeat extraction, input dimension, DL architecture, for CNN-based architectures: number of convolutional layers, number of filters and filter sizes, for RNN-based architectures the number of layers and units in each layer, performance measures.

3.3.4 Analysis of data extracted on DL for the ECG analysis

The number of publications increased each year significantly as presented in Figure 3.5 (e.g.: in 2017 the total number of identified publications by keywords was 74, whereas in 2018 the number increased to 202), therefore, this is the first time that the following analysis has been performed on the information extracted from the selected articles:

1. Bibliometric analysis;

   Bibliometric networks, such as networks of citation relations between publications, networks of co-authorship or networks of co-occurrence relations between keywords represent an exploratory method of the impact of a certain field of research, the impact of certain researchers, or they help to identify impactful papers in a specific field of research [119]. VOSviewer software [32] was used in this review to generate bibliographic coupling networks of publications and sources of the final shortlisted papers included in the review. Moreover, VOSviewer was used to build a co-occurrence map of the authors’ keywords co-occurrence, which can help to investigate the recurrent applications and methods in the shortlisted publications. The visualisation method for the analysis of bibliographic data implemented in VOSviewer is a distance-based method, in which the distance between two items reflects the strength of the relations between the items [32].

2. Synthesis of the most recurrent public ECG datasets;

3. Synthesis of the recurrent DL architectures used for the ECG analysis;

4. Analysis of the DL architecture structures in relation to the input dimension;
5. Synthesis of the common ECG pre-processing steps, with a focus on filtering, normalisation methods or whether the input represents a heartbeat or an extracted ECG segment of a specific window of time;

6. Synthesis of different DL validation methods: intra-subject and inter-subject split for training and testing the models;

### 3.3.5 Results

According to the literature search strategy described above, the total number of papers identified in PubMed (94) and Scopus (360) databases was 454. After removing 81 duplicates, and 39 conference proceedings books, the remaining 334 papers were screened by title and abstract out of which 156 titles were excluded as they did not meet the inclusion/exclusion criteria. Therefore, the remaining 178 titles were read in full-text, after which an additional 72 titles were excluded due to inclusion criteria violation. Finally, 106 studies were shortlisted for this review; a flowchart of the paper selection process is presented in Figure 3.2.

#### 3.3.5.1 Public ECG data.

A summary of major public ECG datasets used in the surveyed publications is presented in Table 3.1. PhysioBank datasets [3], available at: https://physionet.org/, gathers over 60 databases of physiological signals that are freely available and represents the predilect used source of ECG data used in the surveyed studies. Datasets that are part of the PhysioBank database were marked with ‘*’ in Table 3.1.
<table>
<thead>
<tr>
<th>ID</th>
<th>Dataset Description</th>
<th>Dataset Abbreviation</th>
<th>Number of recordings</th>
<th>Recordings Type or Duration</th>
<th>Type of recording</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MIT-BIH arrhythmia*</td>
<td>MITDB</td>
<td>48</td>
<td>30 min excerpts, 2 channel ambulatory ECG</td>
<td>360Hz</td>
<td>ECG classification</td>
</tr>
<tr>
<td>2</td>
<td>MIT-BIH Supraventricular Arrhythmia Database</td>
<td>MITSVDB</td>
<td>78</td>
<td>30 min excerpts, 2 channel ambulatory ECG</td>
<td>128Hz</td>
<td>supraventricular arrhythmia detection</td>
</tr>
<tr>
<td>3</td>
<td>MIT-BIH Malignant Ventricular Arrhythmia Database</td>
<td>MITMVADB</td>
<td>22</td>
<td>30 min excerpts, 2 channel ambulatory ECG</td>
<td>250Hz</td>
<td>ventricular arrhythmia detection</td>
</tr>
<tr>
<td>4</td>
<td>MIT-BIH atrial fibrillation database*</td>
<td>MITAFDB</td>
<td>25</td>
<td>2 channel ambulatory ECG</td>
<td>250Hz</td>
<td>atrial fibrillation detection</td>
</tr>
<tr>
<td>5</td>
<td>PTB Diagnostic ECG Database*</td>
<td>PTB</td>
<td>290 recordings</td>
<td>38-105s, 15 leads 1000 Hz</td>
<td>various conditions</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Physionet MIMIC*</td>
<td>MIMIC</td>
<td>93</td>
<td>1 to 5 recording sessions/subject</td>
<td>multimodal signals &amp; periodic measurements from bedside monitor</td>
<td>ICU patients</td>
</tr>
<tr>
<td>7</td>
<td>Physionet 2014 challenge*</td>
<td>PHYSIO2014</td>
<td>200 for train, 210 for test</td>
<td>multi-parameter recordings (4 to 8 signals/subject)</td>
<td>∼20-40h ECG/500Hz, healthy and a wide range of problems</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Fetal ECG synthetic database*</td>
<td>FECGSYNDB</td>
<td>10 simulated pregnancies, 32 abdominal channels</td>
<td>5 minutes</td>
<td>250 Hz</td>
<td>large database of simulated adult and non-invasive fetal ECG (NI-FECG) signals</td>
</tr>
<tr>
<td>9</td>
<td>DalsaY</td>
<td>DAISY</td>
<td>2500 subjects</td>
<td>15s</td>
<td>8 channels</td>
<td>cutaneous potential recordings</td>
</tr>
<tr>
<td>10</td>
<td>Physionet 2017 challenge*</td>
<td>PHYSIO2017</td>
<td>12186 ECG recordings, 5008 ECG for training</td>
<td>300Hz</td>
<td>atrial fibrillation classification</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>The MIT-BIH Normal Sinus Rhythm Database</td>
<td>NSRDB</td>
<td>18 long term ECG</td>
<td>30 recordings, 12 lead ECG</td>
<td>128 Hz</td>
<td>healthy Sinus rhythm subjects</td>
</tr>
<tr>
<td>12</td>
<td>Physionet 2011 challenge*</td>
<td>PHYSIO2011</td>
<td>10s, 500 recordings</td>
<td>500 Hz</td>
<td>signal noise detection</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>MrOS Sleep Study [1,20]</td>
<td>MrOS</td>
<td>2517 subjects</td>
<td>30 recording sessions, 512 Hz, overnight polysomnography</td>
<td>sleep apnea detection</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>BIDMC Congestive Heart Failure*</td>
<td>BIDMC CHF</td>
<td>15 patient/long term recordings</td>
<td>250 Hz</td>
<td>severe congestive heart failure</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Dataset</td>
<td>Abreviations</td>
<td>Number of subjects or recordings</td>
<td>Type of recording</td>
<td>Task</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>--------------</td>
<td>----------------------------------</td>
<td>-------------------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Fantasia*</td>
<td>FANTASIA</td>
<td>40 recordings</td>
<td>∼120 minutes ECG</td>
<td>affective detection</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>AMIGOS</td>
<td>AMIGOS</td>
<td>∼40 participants</td>
<td>∼120 minutes ECG</td>
<td>paroxysmal atrial fibrillation detection</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>The PAF prediction database*</td>
<td>PAF</td>
<td>50 subjects</td>
<td>∼120 minutes ECG</td>
<td>arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Geriatric University of Verona</td>
<td>CUBDB</td>
<td>35 ECG</td>
<td>∼50 minutes</td>
<td>arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Drug ECG Challenge 2013</td>
<td>ED</td>
<td>50 subjects</td>
<td>∼30 minutes</td>
<td>arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>ECG ID*</td>
<td>ID</td>
<td>310 recordings</td>
<td>∼20s</td>
<td>biometrics</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>The MIT-BIH Noise Stress Test Database*</td>
<td>NSTDB</td>
<td>12 recordings</td>
<td>3 types of noise: baseline wander, muscle artifact, electrode motion artifact</td>
<td>noise detection</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Sin cerebrovascular disease *</td>
<td>CHINA</td>
<td>∼60 minutes</td>
<td>baseline-walker stroke, muscle artifact, electrode-motion artifact</td>
<td>noise detection</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>The PAF Prediction Database*</td>
<td>PAF</td>
<td>∼120 minutes</td>
<td>250 Hz</td>
<td>arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>The MIT-BIH Noise Stress Test Database*</td>
<td>QTDB</td>
<td>30 records</td>
<td>350 Hz</td>
<td>arrhythmia detection &amp; P wave detection</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Check Your Biosignals Here initiative*</td>
<td>CYBHI</td>
<td>∼2 minutes</td>
<td>1kHz</td>
<td>biometrics</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>The American Heart Association Database</td>
<td>AHADB</td>
<td>154 records</td>
<td>∼35 minutes</td>
<td>ventricular arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>DEEPQ</td>
<td>DEEPQ</td>
<td>299 patients</td>
<td>∼5 minutes</td>
<td>arrhythmia detection</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>Chinese Cardivascular Disease Database</td>
<td>CCDD</td>
<td>∼200,000 short term ECGs</td>
<td>∼10-30s</td>
<td>different diagnostics</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Multicenter OHCA</td>
<td>OHCA</td>
<td>10s, 500 Hz</td>
<td>5 rhythms</td>
<td>different diagnostics</td>
<td></td>
</tr>
</tbody>
</table>
Figure 3.3 shows the bibliographic coupling network of the 106 shortlisted publications. Two publications are bibliographically coupled if there is a third publication that is cited by both publications. Therefore the larger the number of references two publications have in common, the stronger the bibliographic relation and the nodes are closer to each other in the map. The network presented in Figure 3.3 was obtained for the publications that shared at least 4 citations. The colour of the nodes denotes the year of the publications (more recent were depicted with lighter colours), and the size of the nodes denotes the number of citations of a certain publication. The visualisation of the bibliographic coupling map as presented in Figure 3.3, highlights certain publications that were highly cited, such as: [130]–[132] therefore, such visualisations might provide a good indication of the highly influential publications in the field. It should be noted that the two publications having the highest number of citations were published in 2017, thus it is expected that more recent publications would have less number of citations. Figure 3.4 presents a view of the recurrent publication sources. Again, colours indicate the publication year and close nodes had a stronger bibliographic relation, meaning that they shared more references which were published in the same journal/conference. The size of the nodes represents the number of citations for a certain source. Inspecting the network it was clear that 2 journals: 'Information Sciences' (230 citations, 4 publications) and 'Computers in Biology and Medicine' (175 citations, 5 documents) were the most influential concerning the total number of citations. Interestingly, the network also revealed that the preferred journals changed during the years, the more recent publications (starting from 2018) regarding DL for ECG analysis were published in 'Nature Medicine', 'IEEE Access', 'Biomedical Signal Processing and Control', 'Expert Systems with Applications' or 'Pattern Recognition Letters' and less in 'Information Sciences'.

Figures 3.7 and 3.8 present the authors’ keywords co-occurrence map selected from the final papers included in the review. Different colours indicate the cluster to which a keyword was assigned and the distance between two keywords reflects the strength of the relation between those keywords. The clustering technique and the density mapping algorithms are described in more detail in Van Eck and Waltman [32]. The keywords that were used for the original papers search (DL, CNN, RNN, ECG) were excluded from the list of considered keywords as these terms would have clearly dominated the keywords map. Figures 3.7 and 3.8 reveal that certain clusters corresponded to different applications of DL for the ECG signal, such as: ECG classification, myocardial infarction, arrhythmia detection, biometrics, internet of things, emotion recognition, heart pressure inference or corresponded to different
Figure 3.3: Bibliographic coupling map of the selected publications with respect to the referenced documents.

Figure 3.4: Bibliographic coupling map of the selected publications with respect to the referenced sources.
ML methods used for the ECG analysis: LSTM, Convolutional Recurrent Neural Network (CRNN), Gated Recurrent Unit (GRU) (a type of RNN). Therefore, the bibliographic mapping identified the key research areas of the applications of DL using the ECG signal as they are further presented in Figure 3.6.

3.3.5.3 Recurrent DL architectures and their structure

The histogram of the general DL methods employed in the shortlisted papers is presented in Figure 3.9 and Table 3.2. It is therefore clear that the preferred DL architecture was mainly based on CNNs and their variations. As emphasised in section 2.4, the choice for employing CNN based networks was motivated by their ability to learn hierarchical set of features, therefore, for the CNN’s ability to automatically learn different characteristics of the input data for a specific target. Moreover, as shown in Figure 3.6, ECG classification (44 studies out of 106) was one of the most investigated problems using DL. ECG classification focuses on the automatic identification of heartbeats of different nature: arrhythmia, atrial/ventricular fibrillation, ectopic heartbeats. In this context, as also highlighted in Figure 3.10, CNN based methods were mostly employed for the ECG classification task, thus, for the automatic detection of the changes in ECG heartbeat morphology induced by different cardiac conditions. In fact, CNN based methods were the most recurrent method
Figure 3.6: Histogram of DL applications using ECG signal, presented in the selected publications using the inclusion/exclusion criteria. For the articles that addressed multiple applications, each application is considered in the histogram.
Figure 3.7: Author’s keywords co-occurrence in selected papers, included in the review as distance-based map. Colours indicate the cluster to which a keyword was assigned, also the distance between two items reflects the strength of the relation between them.

employed for the majority of the applications, with a few exceptions, as highlighted in Figure 3.10: ECG compression, ECG reconstruction, HRV features inference, blood pressure estimation, sleep-disordered breathing and power-line suppression.

Figure 3.11 illustrates the relation between the number of convolutional (conv) layers (x axis), the number of filters in each convolutional layer (rows in the left figure), the filters’ sizes (rows in the right figure) and the input size (y axis). It becomes clear from Figure 3.11 that there was a high variation in terms of the used CNN architectures: the input sizes varied from 16 samples to 38400 samples, the number of convolutional layers varied from 1 to 34, the number of employed filters varied from 2 to more than 120 and finally the filter sizes varied from 1 to more than 100. Although CNN based models were the preferred method employed for ECG analysis, the high variations highlighted in Figure 3.11 draw special attention about the need of proposing state of the art CNN hyperparameters, that should provide good initial values for the number of filters, layers, size of filters in connection to the input dimension, task to be solved.

A few patterns could be observed in terms of the number of the used convolutional
filters: either the number of filters were kept constant in all convolutional layers (illustrated by the rows of the same colour in figure on the left) or the number of filters increased in the deeper convolutional layers (darker colours in deeper layers). In general, it can be observed that the filter sizes were kept constant in all layers more often than the number of filters. Moreover, in contrast to the number of filters, the filter sizes tended to decrease in the deeper convolutional layers. The majority CNN architectures (specifically, 101) did not employ more than 12 convolutional layers, with a few exceptions that employed deeper CNNs, but contained either skip connections or different residual blocks. Figure 3.12a presents the correlation between the input size (i.e. time series length < 4000 samples) and the number of convolutional layers, confirming no linear correlation ($r = 0.14$ and $p = 0.15$) between the two variables. Inspecting the bivariate kernel density estimate between the input size and the number of convolutional layers, shown in Figure 3.12b, it can be observed that the highest density is concentrated in the region bounded by an input size of < 1000 samples and 2 to 4 convolutional layers.
Table 3.2: Summary of the surveyed articles by the employed DL method

<table>
<thead>
<tr>
<th>Method</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>[133], [134], [135], [136], [137], [138]</td>
</tr>
<tr>
<td>AE+LSTM</td>
<td>[139]</td>
</tr>
<tr>
<td>ANN+LSTM</td>
<td>[140]</td>
</tr>
<tr>
<td>CNN</td>
<td>[141], [142], [143], [144], [145], [146], [147],</td>
</tr>
<tr>
<td></td>
<td>[148], [149], [150], [151], [152], [153], [154],</td>
</tr>
<tr>
<td></td>
<td>[155], [156], [157], [158], [159], [160], [161],</td>
</tr>
<tr>
<td></td>
<td>[162], [163], [164], [165], [166], [167], [168],</td>
</tr>
<tr>
<td></td>
<td>[169], [170], [171], [172], [173], [174],</td>
</tr>
<tr>
<td></td>
<td>[175], [176], [177], [178], [179], [180], [181],</td>
</tr>
<tr>
<td></td>
<td>[182], [183], [184], [185], [186], [187], [188],</td>
</tr>
<tr>
<td></td>
<td>[189], [190], [191], [192], [193], [194], [195],</td>
</tr>
<tr>
<td></td>
<td>[196], [197], [198], [199], [200],</td>
</tr>
<tr>
<td></td>
<td>[201], [202]</td>
</tr>
<tr>
<td>CNN ensembles</td>
<td>[203]</td>
</tr>
<tr>
<td>CNN+active learning</td>
<td>[204]</td>
</tr>
<tr>
<td>CNN+features</td>
<td>[205], [206], [207]</td>
</tr>
<tr>
<td>CNN+lap*</td>
<td>[208]</td>
</tr>
<tr>
<td>CNN+LSTM ensembles+rules</td>
<td>[209]</td>
</tr>
<tr>
<td>CNN+RNN</td>
<td>[148], [210], [211], [212], [213], [214], [215],</td>
</tr>
<tr>
<td></td>
<td>[176], [216], [217], [191], [218], [219]</td>
</tr>
<tr>
<td>CNN+RNN+siamese</td>
<td>[220]</td>
</tr>
<tr>
<td>CNN+spp*</td>
<td>[221]</td>
</tr>
<tr>
<td>DBN</td>
<td>[222], [223]</td>
</tr>
<tr>
<td>DBN+features</td>
<td>[224]</td>
</tr>
<tr>
<td>GAN</td>
<td>[225]</td>
</tr>
<tr>
<td>RNN</td>
<td>[44], [226], [227], [176], [191], [228], [229],</td>
</tr>
<tr>
<td></td>
<td>[230], [231], [232], [233]</td>
</tr>
<tr>
<td>RNN ensembles</td>
<td>[234]</td>
</tr>
</tbody>
</table>

*lap - Lead asymmetric pooling
*spp - Spatial Pyramid Pooling

3.3.5.4 Recurrent ECG pre-processing steps

1. Denoising. ECG recordings are highly susceptible to corrupting by noise, the frequent noise sources include [27]:

   - **Power Line Interference (PLI)** - caused by the alternating current of the acquisition equipment, usually introduces high-frequency noise (50-60Hz).
   - **Baseline Wander (BW)** - caused by the breathing movement, reflects on the acquired ECG as a low-frequency (below 1 Hz) wave of the recorded baseline signal.
Figure 3.9: Histogram of the surveyed articles per DL architectures. The acronyms for the architectures are generic, therefore they can denote various sub-types/customisations of the base DL architecture. For the articles that employed multiple DL methods, each method was separately considered in the histogram.

- electromyographic interference - the ECG recording devices can also capture the electrical activity generated by the muscles in the body, resulting usually in high-frequency, high-amplitude short-term bursts.
- electrodes movement - ECG artifacts can be caused by the movement of the user, that determines, in turn, the movement of the electrodes.
- pacemaker interference - signals from the artificial pacemakers are captured along with the ECG signal, usually introducing short spikes before the S wave.
- inversion of the lead - due to misplacement of the electrodes.
- other interference components

Therefore the ECG preprocessing steps usually include different types of filtering to reduce the noise introduced by PLI, BW or muscle noise. Filters such as bandpass (1-40Hz), lowpass, highpass and notch are the most frequent options, as they require low computational cost and they are easy to use. More recently, the Discrete Wavelet Transform (DWT) was proposed as a denoising method for the ECG signal [235], [236]. The method is based on computing the DWT of a signal, passing this transform through a threshold, which removes the coefficients below a certain value, then computing the inverse DWT. Sixty-five studies out of the total 106 employed a filtering method, the majority of the studies used wavelet filtering (23) and the second most used filtering method was the bandpass filter.
Figure 3.10: Stacked histogram of the surveyed articles per application and the employed DL architectures. The CNN and AE architecture variations were collapsed to CNN and AE for enhancing the visualisation. For the articles that employed multiple DL methods and applications, each method and application was separately considered in the histogram.
Figure 3.11: Heatmap of the CNNs’ architecture structure employed in the reviewed studies. The x axis represents the number of conv layers, each row in the heatmap represents the number of filters (left) and the sizes of the filters (right) in each reviewed CNN architecture. For the publications that employed more than one architecture, each CNN architecture was considered, therefore the total number of reviewed architectures is higher (107) than the total number of reported studies that employed CNN based methods (65). (first part)
Figure 3.11: cont.
Figure 3.12: Correlations and kernel density estimate plots for inspecting the relation between the input size versus the number of convolutional layers. The low correlation coefficient ($r = 0.14$) and the $p$ value $> 0.01$ confirms that there is no linear correlation between the two variables.

More information about the filtering methods used in each of the reviewed studies can be found in the Table A.0 in the appendix. Thirty-nine studies did not employ any filtering, the motivation was that DL based methods are not that sensitive to noise as the classical ML methods that use feature extraction. Moreover, different DL methods have been recently proposed for signal denoising, that were applied successfully for ECG denoising [226], [231]. These, along with methods for data augmentation, may result in better alternatives to current traditional filtering methods.

2. **Heartbeat detection.** Following the filtering stage, another common ECG preprocessing step involves the signal segmentation, which limits the signal span, therefore determines the input time series dimension. Two different strategies are commonly employed for ECG segmentation:

(a) Fiducial methods (i.e. requiring a heartbeat extraction step)

(b) Non-fiducial methods that use a sliding window to slice (with or without overlap) a longer ECG signal into smaller, equal-length chunks

Fiducial methods require an additional step, that involves a QRS detector algorithm for the annotation of the P, Q, R, and T ECG waves. Depending on the segmen-
Segmentation strategy, the ECG can be segmented into individual heartbeats, using just the R peak location and predefined windows of time before and after the annotated R peak. This method, however, can result in truncated heartbeats (incomplete) or in ECG segments that capture parts of multiple heartbeats. To overcome this problem, P and T ECG waves can be used to determine the exact length of the heartbeat. However, the disadvantage of extracting the heartbeats using the P, R and T fiducial points is that the extracted heartbeats will have different lengths. Usually, these methods are followed by a resampling step, as most of the DL based methods require a fixed-size input. Moreover, the full heartbeat annotation, therefore the determination of the P and T waves is not a trivial task, that becomes even more challenging when dealing with noisy ECG signals recorded with wearable devices in free-living conditions. Reliable P and T wave detection is more difficult than QRS complex detection for several reasons, including low amplitudes, low SNR, amplitude and morphology variability, and possible overlapping of the P wave and the T wave [237].

Several different QRS detector algorithms including thresholding, neural networks [238], hidden Markov models [239], matched filters [240], syntactic methods [241], zero-crossing [242], and singularity techniques [243] have been proposed in the literature in the recent years, a comprehensive review of these algorithms can be found in [244]. The most popular method, that also became the benchmarking method for QRS detection was proposed by Pan and Tompkins [245]. The QRS detection algorithm employed in the case studies presented in this thesis is based on a method proposed by Yazdani et al. [246], [247]. The method aims at proposing a simple preprocessing tool that suppresses perturbations and enhances the R wave in the ECG signal. It involves the following steps:

(a) Compute signal energies from short- and long-term ECG samples using sliding windows, given by equation 3.1 [246].

\[
Coeff(n) = \frac{\sum_{i=n-s_{win}}^{n+s_{win}} (Sig(i))^2}{\sum_{j=n-l_{win}}^{n+l_{win}} (Sig(j))^2} \tag{3.1}
\]

where \(n\) represents the \(n^{th}\) sample of the signal, \(s_{win}\) and \(l_{win}\) are half of the length of short and long sliding windows

(b) Normalise the computed coefficient to have values between [0, 1]

(c) The enhanced ECG is calculated by multiplying the coefficient signal and the original ECG
Finally, QRS complexes can be identified as they have higher energy in comparison to P, T waves.

The number of reviewed studies that employed fiducial-based methods (therefore that extracted one or more heartbeats) using different QRS detector algorithms was 59 in comparison to 47 studies that segmented the ECG using a sliding window of different sizes (usually a couple of seconds of data). The segmentation method determined the input dimension, therefore the high variations in the input size were the result of different segmentation methods.

3. Normalisation. As previously mentioned the ECG signal varies over time in both amplitude and DC offset due to several factors: acquisition equipment, breathing, muscle interference and HRV. Moreover, the latter determines changes in the length of the heartbeat. To account for this variability, half of the reviewed studies (i.e. 53) employed different methods for amplitude and time normalisation of the segmented ECG signals. Regarding the amplitude normalisation, there were 3 main methods employed:

- Min-Max normalisation, which normalises the input time series to values between 0 and 1 using equation: 3.2, a few example studies include [137], [178], [184], [204], [217], [221]:

\[ z_i(t) = \frac{x_i(t) - \min(x_i)}{\max(x_i) - \min(x_i)} \] (3.2)

- Z-normalisation, given by equation 3.3, a few example studies include: [130], [152], [154], [189], [196], [199], [202], [210]:

\[ z_i(t) = \frac{x_i(t) - \text{mean}(x_i)}{\text{std}(x_i)} \] (3.3)

- Normalisation between [-1, 1], using equation 3.4, a few example studies include: [132], [139], [147], [205], [227], [231]:

\[ z_i(t) = 2 \times \frac{x_i(t) - \min(x_i)}{\max(x_i) - \min(x_i)} - 1 \] (3.4)

More details about the normalisation methods used in the reviewed studies can be found in Table A.0 in the appendix.

Time normalisation methods aim to reduce the variations in the ECG morphology induced by the HRV. The proposed methods aim to reduce the segmented heartbeat
to a predefined length, most commonly through resampling. However, the downside is that the R peak annotation is not adequate to employ such methods, usually the end of the heartbeat (T wave) is also required. An interesting approach was proposed by Kachuee et al. [178] in which the heartbeats were segmented based on consecutive RR intervals and the final dimension of the segmented heartbeats was 1.2RR padded with zero to obtain equal lengths. Another approach was proposed by Li et al. in [221], where each heartbeat was segmented using the R peak annotation of the previous ($R_1$), current heartbeat ($R_2$) and following ($R_3$) heartbeats, using the formula: \(0.45R_1R_2 + 0.55R_2R_3\). The segmented heartbeat was used in the experiments as both resampled for equal size input and as variable length input.

Similar approaches were used in [170], [173], [213]

4. Outlier removal. Outlier detection is applied to discard ECG segments that do not represent heartbeats or they represent noisy ECG segments resulted from movement, impedance artefacts or contact loss [27]. Certain software tools such as the Kubios HRV [248] have a built-in QRS detector and embedded algorithms for RR interval artifacts correction. Kubios provides two RR correction algorithms that can identify missed, extra or ectopic heartbeats. One method is a threshold-based method in which the artifacts and ectopic heartbeats are simply corrected by comparing every RR interval value with a local average interval. The second artifact correction method available in Kubios uses the RR interval time series and a time varying threshold that is estimated from the time varying distribution of the RR series. The downside of Kubios is that so far it cannot be used without the user interface using a scripting language, therefore the ECG recordings need to be loaded in Kubios which can be inconvenient for long-term ECG signals.

Other methods for heartbeat outlier detection include checks for the fiducial points’ amplitudes with respect to the annotated R peak, as in [44]. Moreover, certain studies compared the shape of the considered heartbeat with an average heartbeat shape obtained from a predefined timeframe (usually a couple of seconds preceding the heartbeat). The comparison metric is usually the Euclidean distance of each heartbeat to the mean heartbeat; the heartbeats that exceed a certain threshold are discarded [183]. Other outlier removal methods include morphological checks in terms of amplitude in mV and inter-beat intervals, as proposed in [206]. More recently, Pinto et al. [249] proposed a clustering based algorithm for outlier removal that computes a cross-correlation between candidate heartbeats.
3.4 Conclusions

This chapter presents a review of the recent studies on DL applied to ECG analysis. Diverse and plentiful applications could be identified (32 different applications), which highlights the interest in DL methods applied to time series classification and clustering, and in particular the interest in ECG analysis. The majority of the studies used publicly available datasets, introduced in Table 3.1, however the high number of studies published in recent years involving ECG data, is also motivated by the recent advance in cheap, affordable wearable sensors that can be easily used for data collection. This chapter presents several bibliometric views, characterising the methods, applications, sources and citations of the publications selected for the review. Moreover, the chapter presents detailed aspects of the specific DL modules employed and their structure. CNNs were found to be the predilect architecture, however it has been found that the structure of the networks in terms of number of the layers, number of filters, filter sizes, input size has a high variability. Indeed, throughout this thesis a variable number of CNN layers and filters were used for different applications as a result of a sensitivity analysis aiming to maximise the performance of the model. Moreover, no correlation was found between the number of convolutional layers and the input size. Therefore, the need for clear hyperparameter initialisation values has been identified, future studies can be performed to establish default values for certain hyperparameters.

Future work might include a quantitative analysis of the performance of different DL models and architectures that were employed for the same task on the same dataset. However, to facilitate this comparison, standard training and testing datasets need to be defined, as well as standard validation methods (i.e. intra- or inter- subject).

In the context of the review presented in this Chapter, the remainder of the thesis will cover both novel applications and novel DL architectures proposed for specific case studies. The case study presented in Chapter 4, introduces a novel application (i.e. low glucose detection) of ECG analysis using DL methods. Chapter 5 presents a novel CNN based method for heart failure detection and the results presented in Chapter 6 suggest that the computation of HRV features can be bypassed and the raw ECG signal can be used for case studies that employ HRV as a predictive marker. The proposed DL architectures in chapters 4 and 5 were developed in accordance with the findings and the hyperparameters reported in this review.
Chapter 4

Deep Learning for Low Glucose Detection using ECG Recordings

4.1 Chapter overview

The chapter presents an application of precision medicine and DL methods to a novel dataset and application, that can follow the success DL has had in computer vision field. The aim is to contribute to a better understanding of a condition with a significant social impact; specifically, this chapter presents a novel method for non-invasive, nocturnal low blood glucose level detection. This chapter describes two different experiments both performed on healthy individuals. The first study was carried out in free-living conditions whereas the second case study was carried out in a controlled environment, however over a shorter period. Different DL methods were employed for each case study, the chosen methodology was driven by the data availability. The outcomes of this study showed promising results, opening new research directions.

4.2 Background on glucose levels prediction

There is considerable heterogeneity among different patients or patient groups that trials based on patients cohorts may fail to capture, resulting in inaccurate conclusions about the effectiveness of diagnostics or interventions in individuals [250]–[253]. For example, some drugs have been reported to be effective only in 1 out of 24 patients, because of these inaccuracies [253]. This has triggered unprecedented
interest in personalised medicine (or precision medicine), to inform the design of more effective diagnostics and treatments based on high-quality evidence gathered from individuals, rather than cohorts, considering their personal history, genome, environment, lifestyle, physiology and behaviours. Discussion on how to overcome existing barriers for designing proper precision medicine studies is still ongoing, and overcoming current limitations of traditional trials requires different strategies for therapeutics [253] and diagnostics [254], drugs [255], [256] and medical devices [257]. For the latter, moving towards the paradigm of precision medicine can be even more critical. The effectiveness of medical devices is dependent upon several factors including the environment in which they are utilised (e.g., sterilised or not, temperature, humidity, dust), together with the experience of the operators in using, maintaining, servicing and preparing the medical device [258]. For instance, recent studies demonstrated that even the positioning of applied components (e.g., probes, sensors) in a trial might affect the significance of the results [259]. Therefore, pursuing a precision medicine approach, in trialling AI-based medical technologies requires particular attention in the way the study is designed, executed and analysed [260]–[262].

The study presented in this chapter aims at exploring the efficacy of a personalised AI-based system for the automatic detection of lower levels of glucose in real-life settings. To the best of my knowledge, this is a novel approach providing promising results, overcoming the limitations of previous attempts not based on personalised approaches and not employing AI methods.

Tracking the fluctuations in the blood glucose level is relevant for both healthy individuals and diabetic patients. High glucose levels (hyperglycemia) can result in long-term complications and can damage the kidneys, nerves, blood vessels in the eye and can bring many other complications [263]. Low blood glucose levels (hypoglycemia) may result in acute short-term alterations of health status such as confusion, irritability, palpitations, feeling shaky and sweaty and can even result in severe loss of attention, coma or death [264]. Moreover, hypoglycemia can be particularly dangerous during specific activities requiring great attention (e.g. while driving, performing complicated surgeries). Thus, technologies for non-invasive, continuous monitoring of glucose concentration aimed at early-detecting hypoglycemic events are highly required.

The most diffuse methods for blood glucose testing are performed by analysing a drop of blood resulting from a finger prick. However, this method does not provide continuous data, is invasive, cumbersome, expensive, and it has been demonstrated that it affects patient compliance with glucose measurements [265]. As an alterna-
tive, Continuous Glucose Monitor (CGM)s were developed, they can infer the blood glucose levels in real-time based on the glucose in the interstitial fluid. These devices significantly empowered diabetic patients, but still, they present some limitations that make them unattractive for pre-diabetic patients and diabetics. Specifically, commercially available CGMs can only be worn for a limited number of days, usually between 7 and 14 days, most of the CGMs require finger prick calibration, some studies report that the reliability of CGMs is limited [266]–[268] during low blood glucose level events and how they sample makes them still invasive, although minimally. Moreover, CGMs are quite expensive, which may limit their use for continuous daily glucose monitoring, especially in pre-diabetic patients. However, despite these potential caveats, recent research has revealed that CGM systems could overcome the limitations of self-monitoring of blood glucose (SMBG) using glucometers by providing a complete glucose profile and a detailed history of nocturnal glucose levels. Therefore CGMs are improving glucose control in diabetic patients [268]–[271]. International standards for reliability of the SMBGs have been designed (i.e., ISO 15197:2013), while similar standards for CGMs have not been published. Several metrics have been proposed to characterise the accuracy of the CGMs and one, in particular, has emerged as being the most recurrent measure for the sensor accuracy, which is the mean absolute relative difference (MARD). Different studies reported MARD values of 9.5% to 19% for different CGM sensors [266], [268]–[275], which are close to the values reported for glucometers (5.6% and 20.8%) [276].

A number of non-invasive (e.g., without skin penetration) technologies have been proposed, usually, these are referred to as non-invasive continuous glucose monitors (NI-CGMs). These devices leverage techniques such as Raman spectroscopy [277], fluorescence technology [278], optical coherence tomography [279] and optical polarimetry [280], all aiming to exploit the changes in the chemical and physical tissues properties determined by the glucose variations. Recent reviews [280]–[282] of these devices showed that they are promising, although the underlying technology might be improved, making them more accurate, more comfortable to wear, operate, maintain and calibrate.

Finally, the majority of CGMs’ technologies are not yet developed to combine glucose measurements with other physiological signals or activity measures, which may reflect the subject’s physical and emotional conditions.

The increased number of wearable non-invasive sensors developed for tracking activity or cardiac signals (e.g., ECG) are creating new and unexplored opportunities for the early detection of hypoglycemic events. New strategies have been proposed
to overcome the limitations of currently available, traditional CGM devices such as: combining direct glucose data with physiological parameters to improve the accuracy of the readings (i.e. an enhanced-direct CGM) [283]–[287]; combination of physiological parameters, vital signs, food intake for the estimation or prediction of either glucose levels or hypoglycemia/hyperglycemia events (i.e. indirect CGMs) [288], [289]; use of just the ECG data to detect or predict hypoglycemia (i.e., minimally-invasive indirect CGM) [290]–[293]. The latter approach seems very promising as ECG can be recorded, transmitted and processed quite easily and ECG sensors can be embodied in every day use objects (e.g. a car steering wheel, the backrest of an office chair, or smartwatches). Moreover, ECG-based glucose detection can be more cost-effective and attractive for pre-diabetic individuals or patients suffering from other comorbidities, who may be familiar with ECG monitoring applications both for clinical and consumer (e.g. sport, fitness) applications.

Associations between ECG parameters (e.g., mainly the QT interval duration) and glucose levels have been investigated in both healthy and diabetic subjects [33]–[38] in the recent years. It is known that blood glucose concentration can affect the electrical activity of the heart, although the mechanisms behind these changes are not yet completely understood, according to Lipponen et al. [36]. Two reported mechanisms are hypokalemia and the disruption of the neural regulation system. Hypokalemia increases potassium conductivity in the myocardial tissue resulting in shortened action potentials. This is known to affect the ECG, causing ST depression, biphasic T-wave (first positive, then negative) followed by a positive U-wave [110]. Both hypokalemia and neural regulation are fast, and thus changes in the ECG should be coincident with the occurrence of low blood glucose levels [36]. A third possible cause is that low blood glucose levels affect hormonal secretion, which will determine a delay in the cardiac changes with respect to the onset of hypoglycemia [36].

A variety of methods have been proposed to detect low glucose levels using different combinations of ECG features, including PCA [294], genetic algorithms [293], particle swarm optimization [295] and neural networks [296]. However, none of the studies proposed a model, tool or method to automatically detect low glucose levels using the raw ECG waveform. Commonly, the used ECG features include the QT interval, the RT-amplitude ratio and heart rate (HR) [292]. HR, QT interval, change of HR and change of QT were used as inputs in a system for hypoglycemia detection in type 1 diabetic children based on extreme learning machine methodology in [290]. However, ECG feature extraction suffers from high sensitivity to ECG anomalies such as significant changes in the T-wave morphology (e.g., flat or inverted) as a
reliable measurement of QT is not straightforward [292], [294], [297]. Moreover, the majority of the studies investigating ECG-blood glucose associations have been carried out in a controlled clinical setting and not in real-life conditions. Several studies which investigated the hypoglycemic effects on ECG, recruiting healthy participants, induced low glucose levels using the clamping technique [298], in order to bring the blood glucose concentration to values between 3 mmol/L and 3.5 mmol/L [33], [36], [292], [294]. Also, the studies reported as main limitations the difficulty of handling ECG anomalies (mainly changes in the T-wave morphology) and the small number of participants especially in relation to high differences in individual ECGs [117], [299]. Besides, all these approaches required heavy crafting of data preprocessing and feature extraction, selection and prioritisation.

Recently, significant effort has been spent on exploiting deep neural networks, mainly CNNs and RNNs for time series classification (see chapter 3). Those applications consistently demonstrated improved results over traditional data-mining methods that rely on predefined, manually-extracted features [50]. Consequently, DL applications for physiological time series are growing exponentially, as presented in chapter 3. Among the most successful applications that exploit the ECG signal with DL are the ECG classification, arrhythmia detection, screening for proximal atrial fibrillation or cardiac contractile dysfunction [28], [155], [157], [163], [300], [301].

This chapter presents the results of a pilot project aiming to develop a personalised DL system for automatic nocturnal low glucose level detection in healthy individuals, based only on ECG signals acquired with wearable devices in every day living conditions. This study followed a personalised medicine approach in response to the existing literature that has shown that hypoglycemia detection systems trained on cohorts of participants failed to capture the inter-subject variability in the ECG and the ECG changes induced by hypoglycemia. Finally, the proposed method could aid clinicians visualising the peculiar ECG changes that are the most informative for the automatic detection of low glucose levels in each individual, making the proposed method more transparent.

4.3 Case study 1 - Free-living conditions

4.3.1 Experiment protocol

Eight healthy volunteers that were not taking any medication were monitored during their daily activities without any constraints on diet or lifestyle between 8 and 14 consecutive days. More information about the participants’ demographics and their
Table 4.1: Participants’ demographics, eligible participants highlighted in green.

<table>
<thead>
<tr>
<th>Participant ID</th>
<th>Age</th>
<th>Gender</th>
<th>Weight (kg)</th>
<th>Height (cm)</th>
<th>BMI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>35</td>
<td>M</td>
<td>80</td>
<td>180</td>
<td>24.6</td>
</tr>
<tr>
<td>2</td>
<td>29</td>
<td>F</td>
<td>55</td>
<td>162</td>
<td>20.9</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>M</td>
<td>81</td>
<td>184</td>
<td>23.9</td>
</tr>
<tr>
<td>4</td>
<td>27</td>
<td>F</td>
<td>63</td>
<td>167</td>
<td>21.7</td>
</tr>
<tr>
<td>5 (pre-diabetic)</td>
<td>40</td>
<td>M</td>
<td>64</td>
<td>170</td>
<td>22.1</td>
</tr>
<tr>
<td>6 (pre-diabetic)</td>
<td>56</td>
<td>M</td>
<td>98</td>
<td>168</td>
<td>34.7</td>
</tr>
<tr>
<td>7</td>
<td>58</td>
<td>F</td>
<td>75</td>
<td>158</td>
<td>30</td>
</tr>
<tr>
<td>8</td>
<td>26</td>
<td>M</td>
<td>78</td>
<td>183</td>
<td>23.2</td>
</tr>
</tbody>
</table>

glucose profiles are presented in 4.1. Given the small number of participants included in the study, the age range is quite narrow from 26 to 58 years of age, however I tried to balance the number of males and females with diverse BMIs. No other factors were considered for this analysis such as gender differences, age-related differences or meals/fluid intake. Four participants were excluded from the study, due to a shortage or lack of hypoglycemic events, as defined in the exclusion criteria presented in the following paragraphs. Nominal 24h ECG was recorded with wearable commercial devices (Medtronic Zephyr BioPatchTM HP [112]), with a sampling ECG frequency of 250Hz. The ECG monitor can store up to 3 days of ECG recordings, its battery can last for 36 hours and can be fully charged in less than one hour. Therefore, each volunteer was given two devices and instructed to change them approximately every 24-hours before showering. The Zephyr also records also 3-axis accelerations and a breathing waveform. Based on the raw accelerations an activity parameter was computed and logged by the device at 1Hz, measured in vector magnitude units (VMU), a parameter that was also included in the proposed framework in addition to the ECG signal.

Continuous glucose levels were measured using the FreeStyle Libre Flash glucose monitoring system [302], which measures the interstitial glucose every 15 minutes. Each glucose sensor can be used for up to 2 weeks, also while showering, and according to the producer does not require any calibration with finger pricks. The recent development in CGM technology [268], [303], enabled the use of a factory-calibrated flash glucose monitoring system (FreeStyle Libre) for baseline glucose levels readings in this study, given the real-life requirement. The FreeStyle Libre system is clinically proven to be accurate, stable and consistent over 14 days compared to blood glucose testing without the need for finger-prick calibration [268], [275], [304], [305]. In a clinical study involving 72 type 1 and type 2 diabetic patients, the FreeStyle Libre system achieved 11.4% Mean Absolute Relative Difference (MARD) compared to blood glucose testing and 99.7% of glucose results fell within Zone A and Zone B of
the Consensus Error Grid, when compared against blood glucose testing [268]. A comparative study assessing 17 point-of-care glucose meters, showed that the accuracy varied widely from 5.6% to 20.8% MARD, therefore providing evidence that CGM accuracy is comparable with the accuracy of the point-of-care glucose meters.

Moreover, in July 2018 the Food and Drug Administration (FDA) approved the FreeStyle Libre device, the decision came after Abbott published a clinical trial involving 95 subjects, which found that patients who used the scanner frequently had improved glycemic control and less hypoglycemia, reporting an overall MARD of 10.1% compared to blood glucose testing [270]. Moreover, the CGM readings in this study were used to assess whether the glucose levels dropped below a threshold. Therefore, the CGM readings were used to make a dichotomic decision and not for determining precise glucose values, which enabled us to use CGM readings as a baseline for the ECG classification. Most importantly, to further account for the potential CGM readings inaccuracies, the heartbeats corresponding to a glucose level > hypoglycemia threshold and < hypoglycemia threshold + 0.5% were not considered during training.

During the 24 hours, the ECG sensor was typically removed during showering and during high-intensity activity (usually training/workout) which may cause the electrodes to loosen due to sweat or movement of the sensor which can also introduce extra noise in the ECG recordings. Therefore, the available ECG data are variable for each participant during a 24 hour window of time. Some glucose readings might also be missing as the CGM sensor needs to be scanned at least once every 8 hours, in case of a missing scan, the data that exceeded 8h are not logged. Moreover, the first and the last days of recordings were disregarded from the analysis as studies that investigated the CGM performance showed that the accuracy of the glucose recordings is lowest in the first day and that it also decreases towards the end of the recording period [268].

Furthermore, as mentioned earlier, this study concerns the detection of nocturnal (midnight to 9 AM) low glucose events, although the continuous ECG and glucose recordings were collected almost continuously during a 24h period. There are two main arguments for this choice. Firstly, only in a few participants (two), the recorded glucose levels dropped below the considered low threshold during the daytime, but still the available low events were not enough in order to develop and validate the proposed system. Secondly, it is known that cardiac repolarisation has a circadian cycle that usually lengthens during the night [118], [306]–[308], therefore it is essential to consider whether the ECG changes reflect some circadian physiological alterations or they are indeed induced by lower blood glucose concentrations. Therefore,
Table 4.2: Recommended target blood glucose levels by NICE and ADA

<table>
<thead>
<tr>
<th>Target levels by type</th>
<th>Before meals</th>
<th>After 90 minutes after meals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NICE</td>
<td>ADA</td>
</tr>
<tr>
<td>Non-diabetic</td>
<td>4 to 5.9 mmol/L</td>
<td>3.9 to 5.5 mmol/L</td>
</tr>
<tr>
<td>Type 2 diabetes</td>
<td>4 to 7 mmol/L</td>
<td>4.4 to 7.2 mmol/L</td>
</tr>
<tr>
<td>Type 1 diabetes</td>
<td>4 to 7 mmol/L</td>
<td>4.4 to 7.2 mmol/L</td>
</tr>
</tbody>
</table>

due to the expected ECG circadian changes and the shortage of low glucose events during the day, for the analysis only the data that were recorded during the night were considered. Figure 4.2 illustrates these circadian changes in the ECG heartbeat shape during the day and night by plotting the heartbeats as a colour map over a randomly chosen 24h period for each participant included in the study. It can be observed that the QT interval lengthens during the night in the time domain, a change observed in all the considered participants (Figure 4.2, a-d). To ensure that the low glucose detection model does not capture just the associated circadian ECG changes, the analysis in this study was limited to the nighttime period. The flow chart of the proposed experimental protocol is presented in Figure 4.1.

4.3.1.1 Ethical approval

The study protocol was approved by the Ethics Committee of the University of Warwick, UK, (REGO-2018-2205) and each person enrolled gave written informed consent to participate.

4.3.2 Dataset

ECG excerpts of 15 minutes were annotated as corresponding to normal or low glucose levels according to the CGM readings. Since this study focused on healthy subjects monitored in real life (i.e., no induced low-levels via clamping), lower-glucose level (i.e., LGL) episodes were defined as glucose values lower than 4 mmol/L. A normal glucose level (NGL) was defined as a glucose concentration between 4 mmol/L and 7.5 mmol/L, as per international guidelines: American Diabetes Association (ADA) [309] and National Institute for Clinical Excellence (NICE) found in Table 4.2. Histograms of the glucose values during the recorded period were plotted for each participant, presented in Figure 4.3. In order for a subject to be included in the analysis, at least 10% of their recorded glucose values were expected to be less than the LGL threshold plus a small error of $\approx 0.2$ (to account for the CGM reading error) mmol/L, thus less than 4.2 mmol/L. Moreover, the glucose value that corresponds to the 80th percentile of the recorded glucose values was expected to be less than the higher NGL threshold. Thus, the percentiles condition represents...
Figure 4.1: Flow chart of the experimental protocol, including data collection, data pre-processing and model building.
Figure 4.2: Colour maps of the extracted heartbeats during a 24h period (figure on the left) for each participant (figures a, b, c, d) associated with their corresponding glucose and activity levels (figure on the right).
an additional check that the person is healthy and that the majority of the glucose levels recorded during consecutive days lie between the expected values.

The complete dataset comprises of ECG and glucose recordings for 8 participants that wore the two sensors between 7 and 14 days, as shown in Figure 4.3. Four participants were excluded from the analysis because their glucose levels did not go below the established threshold of 4.2 mmol/L (subjects 5, 6, 7), essentially, they did not experience low glucose events. Subject 8 experienced very few low glucose events, violating the condition that at least 10% of the recorded values should be < 4.2 mmol/L. The small number of low glucose events is expected given the fact the participants enrolled in the study were all healthy individuals. The histograms in Figure 4.3 present a detailed view of the glucose levels recorded for every participant and the glucose values corresponding to the 10th, 30th and 80th percentiles, highlighting again the lack of low glucose events in subjects 5, 6, 7 and 8. Moreover, after being enrolled in this study subjects 5 and 6 were diagnosed as being pre-diabetics, a finding that is also reflected by the high glucose values recorded by the ECG. Thus, the remaining 4 subjects included in the analysis were subjects 1, 2, 3 and 4.

4.3.2.1 CNN based system dataset

The final dataset used for building and testing the CNN based system comprised of a list of ECG heartbeats each having associated 2 additional parameters: an activity level and the corresponding glucose value used as output. To account for the reported average lag time of the FreeStyle Libre system readings, which is known to be approximately 5 minutes [268], each heartbeat was associated with the glucose value that corresponded to the current timestamp of the heartbeat plus 5 minutes. Moreover, the heartbeats that corresponded to glucose levels between 4 and 4.2 mmol/L including the boundaries were not considered during training. This measure ensured that no consecutive heartbeats would be considered as both low and normal, therefore reducing overfitting and accounting for the CGM error close to the chosen threshold for hypoglycemia. For each participant, the recording nights were split into 2 separate datasets for training and testing the model, ensuring that every dataset contained nights with low blood glucose events. An additional validation dataset was created by randomly resampling without replacement 20% of the heartbeats included in the training dataset. The final number of extracted heartbeats for each participant corresponding to the normal and low blood glucose levels is presented in Table 4.3. The total number of recorded heartbeats for the nighttime period is presented in Table 4.4, together with the final number of heartbeats selected for the analysis. It can be observed that just a small number of heartbeats
Figure 4.3: Nighttime (midnight to 9 AM) glucose histograms corresponding to all the available participants. The glucose value of the 10th percentile corresponding to participants 5, 6, 7 and 8 is higher (4.9, 7.1, 5.1 and 4.7 mmol/L) than the expected values of 4.2 mmol/L, thus 4 participants could not be considered for further analysis.
Table 4.3: Number of extracted heartbeats from the nighttime ECG recordings for each eligible participant enrolled in the study and their distribution in the training and testing datasets

<table>
<thead>
<tr>
<th>Subject ID</th>
<th>Normal</th>
<th>Low</th>
<th>Normal</th>
<th>Low</th>
<th>Normal</th>
<th>Low</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>Testing</td>
<td>Training</td>
<td>Testing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>37042</td>
<td>38798</td>
<td>35991</td>
<td>22116</td>
<td>3 (3)</td>
<td>22216</td>
</tr>
<tr>
<td>2</td>
<td>51026</td>
<td>18266</td>
<td>68941</td>
<td>6321</td>
<td>4 (3)</td>
<td>6321</td>
</tr>
<tr>
<td>3</td>
<td>92261</td>
<td>21844</td>
<td>69533</td>
<td>5053</td>
<td>6 (2)</td>
<td>69533</td>
</tr>
<tr>
<td>4</td>
<td>28342</td>
<td>34491</td>
<td>46345</td>
<td>13544</td>
<td>4 (3)</td>
<td>46345</td>
</tr>
<tr>
<td>Average</td>
<td>52168</td>
<td>28350</td>
<td>55203</td>
<td>11784</td>
<td>4.3 (2.8)</td>
<td>55203</td>
</tr>
</tbody>
</table>

Table 4.4: Total number of extracted heartbeats from the nighttime ECG recordings for each eligible participant and the final number of heartbeats selected for the analysis

<table>
<thead>
<tr>
<th>Subjects</th>
<th>Normal recorded</th>
<th>Normal selected (%)</th>
<th>Low recorded</th>
<th>Low selected (%)</th>
<th>Normal recorded</th>
<th>Normal selected (%)</th>
<th>Low recorded</th>
<th>Low selected (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>38743</td>
<td>95.6</td>
<td>39504</td>
<td>98.2</td>
<td>40492</td>
<td>88.8</td>
<td>23193</td>
<td>95.7</td>
</tr>
<tr>
<td>Test</td>
<td>53636</td>
<td>95.1</td>
<td>19710</td>
<td>92.6</td>
<td>72575</td>
<td>94.9</td>
<td>6388</td>
<td>98.9</td>
</tr>
<tr>
<td>3</td>
<td>106184</td>
<td>86.8</td>
<td>23008</td>
<td>94.9</td>
<td>74309</td>
<td>93.1</td>
<td>5367</td>
<td>94.1</td>
</tr>
<tr>
<td>4</td>
<td>34975</td>
<td>81</td>
<td>42998</td>
<td>80.2</td>
<td>59902</td>
<td>77.3</td>
<td>16968</td>
<td>79.8</td>
</tr>
</tbody>
</table>

were removed from the analysis during the preprocessing steps, therefore this would have little impact on the reported performance of the model. The variable number of extracted heartbeats for each of the subjects included in the training and testing datasets is given by different number of recorded nights, the ECG quality and the number of hypoglycemia events. The total number of nights with low glucose events are marked in parenthesis in the Training and Testing columns. When the number of low glucose beats was less than 25% of the number of normal ones during training, the majority class was randomly downsampled without replacement. No other specific methodology (such as oversampling, cost-sensitive learning) was employed for balancing the dataset. The validation dataset was used to monitor the training and to early stop, in case the Area under the ROC curve (AUC) evaluated at every 100 steps did not improve over the next 10 evaluations. The best model as assessed on the validation set was saved during the optimisation process.
Table 4.5: Total number of extracted 5-minute ECG excerpts from the nighttime ECG recordings for each eligible participant enrolled in the study and their distribution in the training and testing datasets.

<table>
<thead>
<tr>
<th>Subject ID</th>
<th>Number of 5-minute ECG excerpts</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training</td>
<td>Testing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Normal Low</td>
<td>Normal Low</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>130 119</td>
<td>118  69</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>275  64</td>
<td>315  22</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>409  86</td>
<td>317  17</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>149 122</td>
<td>217  53</td>
<td></td>
</tr>
</tbody>
</table>

4.3.2.2 CNN + RNN based system dataset

The same recording nights were used for building and testing the CNN + RNN system. Instead of considering the individual heartbeats as inputs, the inputs into the RNN network represent the sequence of the first 200 consecutive heartbeats from a 5-minute non-overlapping ECG excerpt. To ensure that each 5-minute ECG excerpt corresponded to a glucose event: either low or normal, the glucose events that did not last for 5 minutes were filtered out. A low glucose event shorter than 10 minutes was most probably caused by an inaccurate glucose reading (thus most probably an outlier). Moreover, to ensure that the HR in the 5-minute ECG excerpts > 40bpm, only those 5-minutes ECG intervals that contained at least 200 heartbeats were included in the analysis. Similarly, the heartbeats corresponding to glucose values between 4 and 4.2 mmol/L were not considered for training the model (CGM reading + 5% error). Table 4.5 presents the final number of 5-minutes ECG segments that were selected and included in the training and testing datasets.

4.3.3 Data pre-processing

Since this study investigated association between ECG beat morphology and glucose levels, the first step was to isolate each heartbeat. This was achieved by detecting a fiducial point (i.e., the R peak) and then selecting a window of time of 640ms around the fiducial point, in analogy to [301] and accounting for the sampling frequency. However different equipment and sampling frequency were used in [301]. The fiducial point for each heart beat was detected using a QRS detection algorithm, similar to the one proposed in [246]. Since the ECG signals were sampled at 250Hz, a window of time of 640ms was isolated counting 160 ECG samples around the R fiducial point (i.e., 60 samples preceding the R peak and 100 samples following the R peak). Based on the normal resting heart rate i.e between 60 and 100 heartbeats per minute,
the 640ms window of time was chosen after investigating the average length of the heartbeats of the considered subjects in order to capture the P and T waves of the heartbeats. Two parameters logged by the Zephyr BioPatch were used to filter the noisy ECG segments: Heart Rate (HR) confidence and the ECG noise. According to device’s specification, the HR confidence takes into account a worn detection indication and the signal-to-noise ratio of the ECG signal. In the current study ECG excerpts with 100% HR confidence and ECG noise < 0.01 were extracted and considered for the beat extraction and further for the analysis. After the heartbeats were isolated they were z-normalised and downsampled, keeping only the \( k \)th sample (with \( k=3 \), chosen after performing several experiments considering \( k=2, 3 \) and 4).  

As shown by Kiranyaz in [301], downsampling the input heartbeat, reducing the number of samples by 3 did not affect the performance of the CNN model and reduced significantly the training time. Thus, the final length of the heartbeat time series was 53 sample points, which represents the input time series of the CNN.

4.3.4 Proposed DL architecture

4.3.4.1 CNN architecture

The proposed CNN network was implemented in TensorFlow [310], it comprises of 15 convolutional layers with a fixed number of 50 filters in each layer, in agreement with previously published models [157], [195], and one fully connected (FC) layer of 30 neurons. The activity level information that was associated with each heartbeat was also included in the CNN network as an additional neuron in the FC layer, as presented in Figure 4.4. The network was trained from scratch, initialising the weights of the convolutional layer as in [311] using the Xavier initialiser. The sizes of the filters used were kept constant, being set to 3, that represents around 5% of the input time series length (53 samples). The employed loss function was the cross-entropy between the estimated class probabilities and the target classes. The chosen optimiser was AdamOptimizer [99] with an initial learning rate of \( 10^{-4} \). Batch normalisation was employed after each convolution and before the ReLU [312] activation. No pooling operation was used except a 0.5 rate dropout after the fully connected layer. The maximum number of training iterations was set to \( 2.5 \times 10^4 \), which represents at least 45 epochs considering a mini-batch of 200 input beats, for all the participants.
Figure 4.4: Proposed CNN based system illustrating the study objectives. To detect the low glucose levels using the ECG signal, three objectives were set: (OBJ 1) was to build a classifier (using a CNN network) for the low glucose levels detection task. Secondly, the chosen method for performing the classification (i.e. CNN) enables us to investigate further the learned representation of the input heartbeats (OBJ 2), representation (embedding) that can be used in for data visualisation/clustering in lower-dimensional space. The method used for the nonlinear dimension reduction is t-SNE [92]. The third objective (OBJ 3) was to investigate the important regions in the input time series (the heartbeat signal) that contribute the most to the final classification result (Grad-CAM method).

Figure 4.5: CNN architecture hyperparameters search
Due to the high flexibility of the CNN structure and the high number of hyper-parameters, we evaluated a combination of architecture and hyper-parameters in an iterative process, using grid-search and manual tuning performed on the first participant included in the study. Regarding the architecture structure, we searched over the number of convolutional layers (3 to 20), different filter sizes (from 3 to 20) and the number of filters in each convolutional layer (20 to maximum 100). The learning rate was manually tuned in order to achieve a faster convergence; the considered values were $10^{-1}$ to $10^{-5}$. The results presented in this manuscript were obtained on the final CNN architecture that achieved the highest performance on the validation dataset, that also minimised the number of parameters. The results of the performed parameter search are presented in Figure 4.5.

### 4.3.4.2 CNN + RNN architecture

The proposed CNN + RNN based system leverages the representation power of the CNNs and connects the obtained representations to a recurrent neural network in order to also capture temporal dependencies between the input heartbeats as shown in Figure 4.6. Specifically, due to the known exploding or vanishing gradients problems the RNNs, for the recurrent block comprises of LSTM cells. The CNN+RNN model works by passing each input (individual heartbeat, $b_i$) through a feature transformation $\phi_V$ with parameters $V$, which in our case is a CNN network, to obtain a fixed-length vector representation. The outputs of $\phi_V(b_i)$ are then passed into a recurrent sequence learning module (i.e. an LSTM network). The recurrent network in a very general form has parameters W and maps an input $b_t$ and a previous hidden state $h_{t-1}$ to an output $z_t$ and an updated hidden state $h_t$.

The final system is instantiated with a sequential input (the consecutive heartbeats extracted from a 5 minute ECG excerpt) and has a static output generated only at the last sequence step, which is the glucose event associated with every 5-minute interval ($b_1, b_2, ..., b_{200} > y(low/normalglucose)$). To predict a distribution over the outcomes $y$, at time step $t$, the outputs $z_t$ of the sequential model are passed through a linear prediction layer, outputs of which are passed through a softmax function to obtain the final class probabilities.

The CNN module comprises 5 convolutional layers each having 50 filters of size 3. The LSTM module comprises a single LSTM layer with 400 units in each LSTM cell and 200-time steps. The weights’ initialiser for both the CNN and LSTM parameters was the Xavier initialiser and all biases were initialised to 0. The CNN + RNN network was trained end-to-end through backpropagation and it was found that a higher dropout (0.6) was needed to avoid overfitting. The batch size was 30, the
Figure 4.6: Proposed CNN + RNN system for low blood glucose detection over a 5-minute window of time. The individual heartbeats were firstly isolated, then grouped into 5-minute segments. Each considered 5-minute segment was chosen if it contained at least 200 heartbeats. This condition also implies that the glucose event (low/normal) should last for at least 5 minutes, thus each 5-minute ECG segment was associated with a single label: low/normal glucose. Each heartbeat was firstly transformed into a feature representation using a CNN network, a representation that was fed as input to the sequence model (LSTM cells). The outputs of the final RNN are the inputs to a linear layer with a softmax producing a distribution $P$ over the two possible outputs: normal or low glucose values.

initial learning rate was set to $10^{-4}$ and the used optimiser was the AdamOptimizer.

4.3.5 Performance evaluation

The performance measures used for both models’ (CNN and CNN + RNN) assessment were accuracy, sensitivity, specificity, and AUC. Due to the developed personalised models, no cross testing across subjects was performed. In addition, from the clinical perspective, sensitivity is considered more relevant than specificity as it shows how well the event was identified (in our case the low glucose events), thus when comparing different models, sensitivity was considered more important.

When training the model, the inputs of the CNN represent the isolated heartbeats, however, the proposed CNN based model does not account for the sequence of beats in a specific timeframe. In the case of a real-time alarming system, predicting a class for every heartbeat will be undesirable and it might be difficult to follow, instead, generating a prediction every 10 minutes is more feasible and closer to the resolution of the CGM devices. For this reason, the model’s performance in a 10-minute window of time was also evaluated, by taking the majority class of the heartbeat predictions in that specific timeframe. The same voting strategy was also applied to the CNN+RNN model.
4.3.6 Localisation of the contributing ECG heartbeat subsequences

In order to obtain the class-discriminative localisation map in a generic CNN architecture, the Grad-CAM method was employed, as described in [79]. This technique was described in more detail in section 2.4, it implies that the computation of the gradient of $y^c$ with respect to feature maps $A$ (in our case the feature maps of the last convolutional layer) that are global-average-pooled to obtain the weights $\alpha^c_k$ similar to the weights $w^c_k$ computed with CAM method, as described in [78]. The weights $\alpha^c_k$ are given by equation 4.1 and capture the importance of the feature map $k$ for a target class $c$. The Grad-CAM heatmap is obtained as a weighted combination of the feature maps. It has been shown [79] that Grad-CAM is a generalisation of CAM and can be used in conjunction with any CNN architecture with fully-connected layers,

$$\alpha^c_k = \frac{1}{m} \sum_{t=1}^{m} \frac{\partial y^c}{\partial A_k(t)}.$$  (4.1)

4.3.7 Visualisation of the data in a lower dimensional space

A nonlinear dimension reduction method was employed to visualize the data in a lower dimensional space, in particular, t-SNE [92]), described in section 2.4 Therefore, t-SNE was applied to the heart beat embeddings as obtained from the fully connected neurons.

4.3.8 Statistical analysis

A series of key ECG parameters was extracted for all the heartbeats included in the training and test datasets. The extracted parameters include: the amplitude of the Q, R, T waves, the QT interval (measured from peak to peak), the RT amplitude (as a ratio of R-wave and T-wave) and the T wave slope (slope of the line that intersects T wave peak and T wave offset point). As the T wave could not be accurately detected for all the extracted heartbeats, the heartbeats that could not be fully segmented were excluded from the analysis. Furthermore, the number of low and normal heartbeats were balanced, by randomly downsampling the heartbeats corresponding to the majority class. The total number of heartbeats included in the statistical analysis for each subject were (n = 29732, 14276, 40642, 30998) corresponding to (subject 1, subject2, subject 3 and subject 4).

Two non-parametric statistical tests (the condition for normality checked using the Shapiro-Wilk test was violated) were performed to assess both intra- and inter-subject ECG features variability. Therefore, the Mann-Whitney rank test was
conducted to test the changes in the individual ECG features between low and normal glucose levels. To test the changes in the ECG parameters between subjects, a multi-way Kruskal-Wallis H-test was performed for each ECG parameter for low and normal glucose conditions separately. A significant four-way interaction between the four subjects indicated that the ECG feature changed significantly for one or more subjects, without specifically indicating between which subjects the ECG features were significantly different. Therefore, to further investigate the pairwise differences between subjects, a post hoc comparison was performed with a two-way Kruskal-Wallis H-test and Dunn’s test. A p-value $< 0.05$ was accepted as evidence of statistical significance.

### 4.3.9 Programming

The DL models were developed in Python employing different libraries such as TensorFlow, Numpy, Pandas, and trained on an Intel Core i7 processor with 32GB RAM. To speed up the training, the High-Performance Computing facilities, i.e. 4 GPU nodes (each having 2 x NVIDIA Tesla K80 GPU cards) provided by the Centre for Scientific Computing (CSC) at University of Warwick were also used.

### 4.3.10 Results

#### 4.3.10.1 Classification of ECG signals that correspond to normal/low glucose values

The aim of this study was to detect the low glucose levels in healthy individuals based on the ECG signals and actigraphy, recorded continuously during a nominal period of 14 nights for each subject. ECG, actigraphy and CGM were recorded using commercial wearable sensors. A total number of 8 healthy participants were recruited, of which 4 met the inclusion criteria (i.e., at least 2 hypoglycemic events lasting more than 5 minutes in at least 2 nights). In fact, as detailed in section 4.3, 4 participants did not experience any hypoglycemic events during the recording period, which was not surprising as all considered participants were healthy. A CNN network was trained on the isolated heartbeats extracted from the raw ECG signal, as detailed in section 4.3. The proposed system is a person-specific one in which data recorded for a participant during the first days, were used for training the model, which was tested using data from the same subject acquired in the remaining days. However, there were a few exceptions when the occurrence of low glucose events was not balanced during the recording period, so the days considered for training and testing were not consecutive. This should not influence the reported performance.
of the model, the consecutive days condition is not mandatory, it was employed to ensure an automatic, easier split of the data into training and testing datasets. In this study, it was assumed that the cardiac changes occurred in the same time as low blood glucose levels, as explained in Section 4.2, and a 5-minute delay in glucose levels readings introduced by the CGM was taken into consideration, as reported by the manufacturer. Actigraphy was measured using commercial sensors embodied in the body-worn ECG device. Activity levels were estimated from the 3-axis accelerations and computed as $VMU = \sqrt{x^2 + y^2 + z^2}$ where $x$, $y$ and $z$ are the averages of the three-axial accelerations over the previous one second. Thus, the detection algorithm was mathematically formulated as the following:

$$GlucoseLevel = f (ECG\text{beat}, \ Activity\text{level}) .$$ (4.2)

The low glucose detection problem was cast to a classification problem in which the inputs represent the extracted heartbeats together with an additional covariate - the activity level and output – normal/low glucose level. To test the feasibility of heartbeat classification by glucose levels, two different approaches were proposed: a CNN based system as presented in Figure 4.4 and a CNN + RNN system, shown in Figure 4.6, the models described in more detail in section 4.3.4. The reason behind the CNN choice lies in its capability of learning hierarchical, abstract representations of the input space that are relevant to performing specific tasks, as presented in chapter 3. The CNN + RNN model was built considering the RNN’s ability of learning sequences, in this case sequences of consecutive heartbeats, which were supposed important for detecting low glucose events. In this combined model, the CNN module was used for learning the heartbeat representation, while the RNN component was responsible for learning the heartbeats sequence in the considered 5 minute ECG intervals. Following a personalised medicine approach, the two models (CNN, CNN+RNN) were trained from scratch for each participant using a variable number of recording nights out of which at least 2 nights should contain low blood glucose events. The number of final heartbeats included in the training/validation and testing is reported in section 4.3.2.

4.3.10.2 CNN based system

For the CNN based system, two evaluation strategies were considered, one in which all the individual heartbeats were classified corresponding to the test days and a second strategy used was to employ a majority-voting scheme for the heartbeats in a 10-minute window of time. The results are reported in Table 4.6 (a and b) for all
Table 4.6: CNN based model classification results, evaluated for each participant. The individual beat column presents the results for each heartbeat classification, whereas the 10-minute voting column shows the results when taking the majority class corresponding to all heartbeats in a 10-minute window of time.

(a) CNN based system classification results, evaluated on testing data for each participant.

<table>
<thead>
<tr>
<th>Subject id</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
<th>Number of correctly predicted 10 minutes intervals/total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Individual beat</td>
<td>10 min voting</td>
<td>Individual beat</td>
<td>10 min voting</td>
</tr>
<tr>
<td>Subject 1</td>
<td>74.2</td>
<td>78</td>
<td>71.2</td>
<td>77.1</td>
</tr>
<tr>
<td>Subject 2</td>
<td>66.0</td>
<td>79.8</td>
<td>69.5</td>
<td>77.1</td>
</tr>
<tr>
<td>Subject 3</td>
<td>82.2</td>
<td>100</td>
<td>87.4</td>
<td>91.9</td>
</tr>
<tr>
<td>Subject 4</td>
<td>81.1</td>
<td>91.5</td>
<td>76.3</td>
<td>80.5</td>
</tr>
<tr>
<td>Average</td>
<td>75.9±7.4</td>
<td>87.5±10.3</td>
<td>76.1±8.0</td>
<td>81.7±7.0</td>
</tr>
</tbody>
</table>

(b) CNN based system classification results, evaluated on training data for each participant.

<table>
<thead>
<tr>
<th>Subject id</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
<th>Number of correctly predicted 10 minutes intervals/total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Individual beat</td>
<td>10 min voting</td>
<td>Individual beat</td>
<td>10 min voting</td>
</tr>
<tr>
<td>Subject 1</td>
<td>91.3</td>
<td>94.6</td>
<td>79.7</td>
<td>80.4</td>
</tr>
<tr>
<td>Subject 2</td>
<td>93.1</td>
<td>100</td>
<td>87.5</td>
<td>93.2</td>
</tr>
<tr>
<td>Subject 3</td>
<td>97.5</td>
<td>100</td>
<td>88.4</td>
<td>89.7</td>
</tr>
<tr>
<td>Subject 4</td>
<td>75.0</td>
<td>83.5</td>
<td>81.3</td>
<td>85.5</td>
</tr>
<tr>
<td>Average</td>
<td>89.2±9.8</td>
<td>94.5±7.7</td>
<td>84.2±4.3</td>
<td>87.2±5.5</td>
</tr>
</tbody>
</table>

The eligible participants on the testing and training datasets. The results obtained on the training dataset are expected to be better than the ones obtained on the testing dataset as the model was optimised on the training dataset. Figure 4.7 presents the same results for training (a-h) and testing (i-p), but visualised as predicted events over the night, in direct comparison to the baseline CGM glucose values. This visualisation provides insightful information about the predicted glucose events otherwise disregarded, including: what time certain events occurred, what time the misclassified events occurred, the connection between misclassified events and the considered low glucose threshold, the lag between CGM glucose readings and predictions etc. Moreover, Figure 4.7 reveals the certainty of the model’s predictions, indicated by different colour intensities (dark green/red for most certain prediction), CGM readings uncertainty (grey shadow around the continuous line) and proximity to the glucose level threshold.

As expected, the majority of misclassified heartbeats and the less-certain classifications (light green/orange) occurred when the glucose levels were close to the low
The results presented in Table 4.6 revealed that, even when the number of normal glucose heartbeats greatly exceeded the number of low glucose heartbeats (10 times more), the proposed system still provided a good result, suggesting that DL was resilient to the unbalanced dataset. This was the case for participants 2 and 3 (Table 4.6).

4.3.10.3 CNN+RNN based system

The CNN+RNN system was evaluated on 5-minutes input ECG excerpts. The first extracted 200 heartbeats in the 5 minute ECG signals were considered as input sequences for the CNN, the output representation of the CNN was fed into a stack of RNN cells, which produced the final prediction, as shown in Figure 4.6. A majority voting was performed for 10 minute ECG segments. The voting strategy ensured that the prediction frequency was similar to the resolution of the CGMs which is usually between 5 and 15 minutes, and it aims to correct the annotation of certain isolated ECG heartbeats. Results are reported in Table 4.7 (a and b) for the test and training days, respectively. Figure 4.7, second column, presents the classification results, showing the CNN+RNN model predictions over the analysed timeframe for a sample night for two of the subjects. Prediction plots for all the subjects and nights can be found in appendix B, section B.1.
Table 4.7: CNN+RNN classification results, evaluated for each participant. The 5-minute column presents the results for the 5-minute ECG segments used during training, whereas the 10-minute voting column shows the results when taking the majority class corresponding to all heartbeats in a 10-minute window of time.

(a) CNN+RNN based system classification results, evaluated on testing dataset.

<table>
<thead>
<tr>
<th>Subject id</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
<th>Number of correctly predicted 10 minutes intervals/total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Individual beat</td>
<td>10 min voting</td>
<td>Individual beat</td>
<td>10 min voting</td>
</tr>
<tr>
<td>Subject 1</td>
<td>79.7</td>
<td>80.5</td>
<td>69.4</td>
<td>73.3</td>
</tr>
<tr>
<td>Subject 2</td>
<td>81.8</td>
<td>81.8</td>
<td>82.2</td>
<td>88.0</td>
</tr>
<tr>
<td>Subject 3</td>
<td>82.4</td>
<td>76.5</td>
<td>89.6</td>
<td>94.6</td>
</tr>
<tr>
<td>Subject 4</td>
<td>100</td>
<td>100</td>
<td>81.1</td>
<td>82.0</td>
</tr>
<tr>
<td>Average</td>
<td>86.0±9.4</td>
<td>84.7±10.4</td>
<td>80.6±8.3</td>
<td>84.5±9.0</td>
</tr>
</tbody>
</table>

(b) CNN+RNN based system classification results, evaluated on training dataset.

<table>
<thead>
<tr>
<th>Subject id</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
<th>Number of correctly predicted 10 minutes intervals/total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Individual beat</td>
<td>10 min voting</td>
<td>Individual beat</td>
<td>10 min voting</td>
</tr>
<tr>
<td>Subject 1</td>
<td>74.8</td>
<td>75.2</td>
<td>84.6</td>
<td>88.6</td>
</tr>
<tr>
<td>Subject 2</td>
<td>85.9</td>
<td>84.1</td>
<td>92.8</td>
<td>96.3</td>
</tr>
<tr>
<td>Subject 3</td>
<td>100</td>
<td>100</td>
<td>84.6</td>
<td>89.2</td>
</tr>
<tr>
<td>Subject 4</td>
<td>92.6</td>
<td>90.2</td>
<td>91.9</td>
<td>94.6</td>
</tr>
<tr>
<td>Average</td>
<td>88.3±10.6</td>
<td>87.4±10.4</td>
<td>88.5±4.4</td>
<td>92.2±3.8</td>
</tr>
</tbody>
</table>
(a) Subject 1, CNN, train

(b) Subject 1, CNN+RNN, train

(c) Subject 2, CNN, train

(d) Subject 2, CNN+RNN, train

(e) Subject 3, CNN, train

(f) Subject 3, CNN+RNN, train

(g) Subject 4, CNN, train

(h) Subject 4, CNN+RNN, train
Figure 4.7: Hypoglycemia detection during the night using the heartbeat majority voting in the 10-minute window of time. The black waveform represents the glucose values recorded by the CGM, considered as ground truth glucose level in this study. The grey shaded regions illustrate a ±10% error boundary for the CGM glucose readings [267], [268] as reported in previous studies. The colour of the points indicates the predicted class: red for the predicted low glucose levels and green for the predicted normal glucose levels. The dark colours indicate more certain predictions: dark red points accounted for low glucose predictions with the predicted probability >0.7, while light red accounted for low glucose prediction with predicted probability ≤0.7; dark green accounted for normal glucose prediction with predicted probability >0.7 and light green accounted for normal-glucose prediction with a probability ≤0.7. Images (a)-(h) present the glucose level predictions for a sample training day, and columns (i)-(p) present the glucose predictions for a sample test day.

4.3.10.4 Localisation of the discriminative heartbeat subsequences

Grad-CAM, as presented in Selvaraju et al. [79] allowed visualisation of class-discriminative sequences in the input heartbeat, without requiring modifications to the CNN architecture or retraining. Unlike other approaches for CNN visualisations, the Grad-CAM method achieves localisation in one shot, it only requires a single forward and a partial backward pass per input time series and it showed better performance in qualitative and quantitative evaluations in comparison to other similar methods such as contrastive Marginal Winning Probability (c-MWP) [313] method. Figure 4.8 presents the histograms of the important sample points in the input heartbeats when Grad-CAM was employed. The highlighted subsequences in the input heartbeats were essential information transmitted through the network, enabling the inspection of the class-discriminative information in the input time-series. Therefore, Figure 4.8 illustrates the subsequences in the input heartbeats that the CNN used in making the predictions. As expected, the onset and the offset
of the T wave was mainly highlighted as important in all subjects. The P wave was indicated as especially important, when the amplitude of the P wave was lower for the low glucose beats than for the normal glucose beats, in subjects 3 and 4. Moreover, the QRS onset and offset were marked as important in subjects 2-4.

### 4.3.10.5 Dimensionality reduction – CNN based system.

The assumption that deeper convolutional layers capture higher-level features, thus that the last convolutional layer should contain the most informative features was extensively explored in previous studies [314], even the Grad-CAM method is based on the last feature maps. However, the activation space of the CNN last layer was too high dimensional for visual investigation and human interpretation. Accordingly, a non-linear dimension reduction method was used to visualise the learned embeddings (i.e., the last CNN layer activations) in lower-dimensional space. Specifically, the t-SNE [92] method was employed for dimensionality reduction. In this study, t-SNE was applied to a balanced subset of the heartbeats included in the test dataset (the normal glucose heartbeats were randomly downsampled without replacement to match the number of low glucose heartbeats). Figure 4.9 presents the t-SNE visualisation of the test heartbeats corresponding to one of the subjects (subject 3) applied to the activation space of the CNN’s last layer. In this space, as reported in Figure 4.9, it can be observed that the heartbeats are organised in two clear clusters that correspond to the low and normal glucose levels. Moreover, Figure 4.9b shows that heartbeats corresponding to lower (i.e., dark red) or higher (dark green) glucose values are clustered in smaller regions. This could be interpreted as an inner validation of the method proposed and a demonstration of the discrimination power of the network’s learned features, which evidently allowed the unsupervised separation of the heartbeats in two different groups that are also in agreement with the corresponding glucose value magnitude.
Figure 4.9: t-SNE visualisation of the test heartbeats corresponding to subject 3 in the activation space representation. The red heartbeats correspond to a low glucose level (< 4.0 mmol/L) and the green heartbeats correspond to normal glucose levels. 
a. t-SNE visualisation when the heartbeats are coloured according to the glucose annotation(class) b. t-SNE visualisation using a colour map that shows the glucose value associated with each heartbeat, the darker the colour the higher the glucose value for normal beats and the lower for low glucose beats.

4.3.10.6 Statistical analysis

The Mann-Whitney rank test revealed that for all the extracted ECG features corresponding to low and normal glucose heartbeats from each subject there were significant differences (p-value < 0.01), detailed results presented in the Figure 4.12. The inter-subject statistical tests showed that the ECG features extracted from low and normal glucose heartbeats were also significantly different between subjects, as shown in Figure 4.10 and Figure 4.11. These results are in agreement with previous studies that showed that not only the ECG morphology is highly variable among individuals but also the ECG changes induced by hypoglycemia [33], [36]. Comprehensive results of this analysis can be found in Table 4.8 and Figures 4.11 and 4.10, which confirmed the high intra- and inter-subject ECG variability hypotheses, therefore supporting the necessity of a personalised approach.
Table 4.8: Kruskal-Wallis H-test between pairs of subjects for the extracted ECG features corresponding to Low glucose (a) and Normal glucose (b). *p*-value post hoc column presents the post hoc pairwise test results for multiple comparisons of mean rank sums using Dunn’s test.

(a) Low glucose

<table>
<thead>
<tr>
<th>Group</th>
<th>Q_amp</th>
<th>R_amp</th>
<th>T_amp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H</td>
<td>p-value</td>
<td>p-value</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S2</td>
<td>8285.5</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S3</td>
<td>5390.59</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S4</td>
<td>17489.95</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S2 &lt;-&gt; S3</td>
<td>1922</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S2 &lt;-&gt; S4</td>
<td>1552.74</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S3 &lt;-&gt; S4</td>
<td>10065.08</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
</tbody>
</table>

(b) Normal glucose

<table>
<thead>
<tr>
<th>Group</th>
<th>Q_amp</th>
<th>R_amp</th>
<th>T_amp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>H</td>
<td>p-value</td>
<td>p-value</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S2</td>
<td>13042.9</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S3</td>
<td>18328.62</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S1 &lt;-&gt; S4</td>
<td>1541.51</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S2 &lt;-&gt; S3</td>
<td>4826.29</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S2 &lt;-&gt; S4</td>
<td>11073.53</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
<tr>
<td>S3 &lt;-&gt; S4</td>
<td>12925.27</td>
<td>&lt;0.01</td>
<td>&lt;0.01</td>
</tr>
</tbody>
</table>
Figure 4.8: Identification of the most relevant heartbeat segments for hypoglycemia detection using the Grad-CAM method. The solid lines represent the mean of all the heartbeats that correspond to each subject in the training dataset: green during normal glucose levels, red during hypoglycemic events. The comparison among 4 different subjects highlighted the fact that each subject may have a different ECG waveform during hypoglycemic events, for instance Subjects 1 and 2 present a visibly longer QT interval during hypoglycemic events, different from subjects 3 and 4. The error bands represent the standard deviation of the considered heartbeats. The vertical bars represent the histograms of the sample points that were > 0.9 in the normalised heatmaps obtained from applying Grad-CAM methods on all the training heartbeats.
Figure 4.10: Box plots for the extracted ECG features during Low and Normal glucose levels for every participant. A multi-way Kruskal-Wallis H-test was performed for every ECG parameter for the low and normal glucose condition separately. The only non-significant differences between the groups are indicated in the plot n.s.
Figure 4.11: Point plots for the extracted ECG features during Low and Normal glucose levels for every participant, showing the relationship between the mean of every ECG feature for low and normal glucose levels.
Figure 4.12: Mann-Whitney rank test on the extracted ECG parameters for each subject, figures a-d. All the statistical tests showed significant differences between the groups (low vs normal glucose level).
4.3.11 Discussion

The results of this study have shown that hypoglycemic events can be automatically detected using a few ECG heartbeats recorded with wearable devices in free-living conditions using personalised classifiers based on DL artificial intelligence algorithms. These results confirmed the two hypotheses stated in the Introduction, showing that both heartbeat morphology and the sequence of heartbeats can be used effectively for low glucose detection during the night, in everyday life conditions. These findings are aligned with clinical studies that presented the predilect cardiac changes occurring during low blood glucose levels, in both healthy and diabetic: prolonged QT interval, increase in the R/T amplitude [33], [36], [37], [315].

Moreover, comparing the performance of the personalised DL system described in this study with previous cohort-based attempts supports the hypothesis that AI can be used to foster precision medicine diagnostics. The proposed system could automatically learn patterns in the ECG heartbeat, discriminating between heartbeats recorded during low or normal glucose levels in the same subject. The use of personalised approaches overcame the limits of conventional cohort-based methods accounting for the inter-subject variability in ECG morphology.

As emphasised in section 4.3.4, the analysis was restricted to nighttime recordings, where detection of hypoglycemic events is more useful and also to minimise the influence the circadian rhythm of the cardiac repolarisation that usually lengthens during the night, as shown in Figure 4.2, [308]. These results demonstrate that the proposed CNN based system could accurately detect 10-minute long low glucose events with high specificity (average 81.7%) and sensitivity (average 87.5%) in comparison to previous attempts detailed below, as presented in Table 4.6. These results advance the idea that no manually extracted features are required to perform this task since DL methods (i.e. CNN) are able to learn automatically highly discriminative features from the raw ECG signals. This is extremely important in the proposed task since feature-based methods are highly dependent upon the correct heartbeat segmentation, where a precise determination of the QT interval would require ECG recording with a high signal-noise ratio that is difficult to measure in real-life and can only be obtained in controlled environments [294]. Thus, the method presented in the current study seems to be particularly useful for real-life settings.

In addition, extra information regarding the sequence of the heartbeats can be captured and presented by combining a CNN with an RNN. The CNN was used to transform the input ECG beats into embeddings that were further aggregated over time by an RNN cell. The predictions for this setup were generated every
5-minutes as the input to the CNN + RNN were the extracted heartbeats in a 5-minute ECG excerpt. The results over 10 minute showed that the CNN + RNN model performed on average better than the simple CNN based system, considering (Table 4.7) sensitivity (average 84.7%), specificity (average 84.5%), average accuracy for CNN was 82.4 % vs. 85.7% for the combined model and from the visual inspection of the prediction plots presented in Figure 4.7. Filtering out the glucose events shorter than 5 minutes might also contribute to the improved performance over the CNN-based system. Disregarding low glucose events shorter than 5 minutes is in agreement with findings that a low glucose event should last for 10 minutes to be considered a true hypoglycemic event and that very short falls in glucose levels do not reveal the related changes in the ECG [36], [316] signal. Moreover, from the visual inspection of the predictions during the night, it can be observed that the regions affected by higher uncertainty (i.e. closer to the threshold of 4 mmol/L) were better classified by the CNN + RNN model, which therefore was considered more resilient.

To the best of my knowledge, this is the first study proposing a personalised system to detect low glucose levels in real-life settings, using the raw ECG signal. Thus, a direct comparison of the obtained results with existing literature is not straightforward. Other studies attempted to detect hypoglycemia through non-invasive monitoring using features extracted from the ECG signal. Studies co-authored by Prof. Hung T. Nguyen [290], [291], [293], [295], [296], [317], [318] involved nocturnal hypoglycemia detection in 15 type 1 diabetic children using different machine learning techniques (extreme learning [290], hybrid swarm optimization [295], neural networks [296], genetic algorithms [293], and a few others), using as inputs different ECG parameters computed from 5 or 10 minutes ECG excerpts, and achieving interesting sensitivity and specificity. For example, the more recent studies, proposed models based on a neural logic approach [318], obtaining 79.07% sensitivity and 53.64% specificity, deep belief network approach [319], achieving 80% sensitivity and 50% specificity, models based on extreme learning approach, obtaining 78% sensitivity and 60% specificity. As already emphasised, direct performance comparison with those studies is not viable as the proposed model is person-specific, which, in my opinion, explains why the proposed method outperforms the results achieved in previous studies. In fact, as demonstrated in Figure 4.8 and Figure 4.10, individual ECG response to low glucose levels varies significantly among different subjects. This affects the classification performance significantly when trying to build a model that can generalise the discriminative features for different individuals. Also, this study concerned the detection of nocturnal non-induced low glucose levels in healthy
individuals; several clinical studies showed that cardiac changes could have different intensities in healthy, type 1 and type 2 diabetic persons [36].

An advantage of using CNN based systems for heartbeat classification is the possibility of producing visual explanations for the network’s decisions, thus making the CNN more transparent. As shown in section 4.3.4, employing a CNN in conjunction with different techniques such as CAM or Grad-CAM could produce a coarse localisation map by using the gradient of the target class with respect to the feature maps of the last convolutional layer, highlighting the important regions or subsequences in the input time series for making a certain prediction. This was important in order to show to the clinical partners, which segment of the ECG excerpt contained the key information utilised by the proposed AI system. Revealing and explaining how the proposed models reached certain conclusions not only makes the models more transparent, but can also disclose interesting information about the underlying data.

Previous studies observed the heartbeat changes associated with hypoglycemic events, but mainly during hyperinsulinemic clamps, allowing the thresholds for both normal and low glucose levels to be entirely controlled and set to specific values. For instance, Marques at al. [320] considered 3 mmol/l as hypoglycemic level and 5 mmol/l as euglycemic level in type 1 diabetic subjects, Laitinen et al. [33] considered the same limits, but in healthy subjects. The common, statistically significant finding in both studies was the lengthening of the QT interval during hypoglycemia. The results presented in Figure 4.8 can be interpreted in agreement with this finding, as in all 4 subjects the T-wave was coloured as being important for both classification tasks (i.e. detecting low or normal levels). T-wave flattening was found to be another significant characteristic of hypoglycemia in Laitinen et al. [33] and a few other studies [36], [292], [294]. The results presented in this study revealed that the changes in the T wave amplitude were personal. Subjects 3 and 4 presented a mean T-wave amplitude even higher for low glucose heartbeats than for normal glucose heartbeats. The reasons for this finding could be manifold. Firstly, the considered subjects in the current study were healthy and the experiment was carried out in free-living conditions, thus on very sparse occasions, the glucose levels dropped below 3.8 mmol/l. In connection to this, it has been shown that during spontaneous nighttime hypoglycemia in type 1 diabetic patients the cardiac repolarisation changes are not as intense as during induced hypoglycemia [316]. Also, the CGM device used in this study to record the ground truth glucose levels were shown to have an overall absolute error difference of 11.4% against capillary blood glucose reference [268]; thus some of the heartbeats could have been annotated incorrectly due to the error.
Another limitation introduced by the CGM is the glucose reading lag which was reported to be 4.5±4 min, however, in the current study, a fixed 5 minutes were accounted for the reading lag, independent of the time, activity or food intake. Further research could be carried out to investigate whether drops in potassium levels are coincident with spontaneous glucose falls in healthy subjects, as it has been shown that low potassium can also determine the flattening of the T-wave [33], [35], [316]. Interestingly, Figure 4.8 reveals that during low glucose levels, the P-wave is more pronounced in some of the subjects (subjects 1 and 4) and that the P-wave might be important for low glucose detection. This intra- and intra-subject variation in ECG morphology reinforced the idea that there is a clear need for adopting personalised approaches for ECG-based glucose level detections.

The unsupervised clustering of the heartbeats corresponding to low and normal glucose levels using the t-SNE method, presented in Figure 4.9, indicates that the CNN network is capable of automatically learning high-dimensional discriminative features. These results demonstrate that the learned feature space can be used to visualise and organise the input data. Data visualisation techniques such as t-SNE can help to inspect the input data and the model, as the similarity of inputs in the original space (thus of the input heartbeats) is also preserved in the obtained low-dimensional space. Figure 4.9 shows that the learned embedding can separate the heartbeats according to the glucose level. Moreover, the heartbeats corresponding to the glucose extremes: low and high form clear-defined clusters, showing that for heartbeats confidently associated with a certain class that the heartbeats are correctly, further separated.

The statistical tests also confirmed the need for developing personalised hypoglycemia detection systems. Moreover, the results provided additional evidence for the less accurate systems developed in the past, that used a pool of ECG features extracted from a cohort of subjects (15 subjects) to develop different statistical models for hypoglycemia detection. As expected, the results from this statistical analysis showed that the inter-subject ECG features differences were statistically significant. Therefore, accurate hypoglycemia alarming systems based on ECG analysis can be developed using personalised ECG-based representation learning methods. Moreover, the personalised approaches proposed in this study showed significant performance improvement in detecting low glucose events over the previous, non-personalised systems.

Given the compelling performance in detecting nocturnal lower glucose levels in healthy individuals using the ECG signal, it is expected that DL based methods similar to the proposed ones in this study may help advance the under-
standing of electrocardiographic changes induced by glucose level variations. Analysis, as demonstrated here, can lead to a better understanding of the underlying processes that determine specific changes in the input heartbeat during low glucose levels. These results show strong evidence that ECG alterations are feasible when used for building a real-time alarm system for low glucose events that occur during the night. The obtained results demonstrate superior performance in detecting low glucose levels in comparison to other similar studies, although due to protocol differences, the results cannot be directly compared.

The proposed person-specific framework for detecting low glucose levels in healthy subjects may be utilised in real-life applications as it involved a few preprocessing steps, and it did not require any expert annotations or feature engineering. These results showed that leveraging DL methodologies for the analysis of ECG in order to detect low glucose events can open new possibilities to develop innovative alarm technologies that might help individuals, especially diabetic patients, to better control their blood glucose concentrations. Alerting the user in real-time when glucose levels fall below a critical threshold value would facilitate the management of hypoglycemia events and can prevent the development of other severe, life-threatening episodes. Therefore, the proposed system creates the potential for long-term improvements in clinical outcomes, especially in diabetic patients. Moreover, providing personalised insights into heartbeat morphological changes during hypoglycemia might also be utilised as a clinical decision support system.

4.3.12 Case study 1 - Conclusions

The pilot study presented in this chapter must also be seen in the light of its limitations, which also represent important calls for future research avenues. Firstly, additional tests should be performed on a larger population, including diabetic patients, to further validate these results. Secondly, the proposed framework can be easily extended to include other physiological signals that might influence the glucose variation such as activity levels, temperature, skin conductivity or nutrition information that might further improve the performance of the system. For diabetic patients, that use finger pricks to check their actual blood glucose levels, implementing online training techniques is essential, as the system should be able to also learn continuously from new data.
4.4 Case study 2 - Human Metabolism Research Unit (HMRU) conditions

In the previous section 4.3 it has been shown that nocturnal hypoglycemia can be accurately detected in free-living conditions, using the raw ECG signal. Two different personalised DL models were developed and validated on 4 healthy participants. Differently from previous studies [290], [295], [296], [318], [319], the case study presented in section 4.3 showed that by employing DL methods, the feature extraction and selection steps could be bypassed, addressing the heartbeat segmentation problem. Moreover, due to the observed high inter-subject variability of the ECG, the case study presented in section 4.3 showed that person-specific models are required for the accurate hypoglycemia detection.

This case study extends on the previous work, presenting further evidence to support the feasibility of a non-invasive hypoglycemia detection system. This section tested an improved version of the proposed CNN model on an additional 8 healthy individuals during a 36 hour period at the University Hospitals Coventry & Warwickshire in the Human Metabolism Research Unit (HMRU).

Although there is compelling evidence that DL methods can be applied for the analysis of physiological signals (see chapter 3), it has been shown that training a deep neural network to directly optimise a supervised objective, starting from a random initialised set of parameters does not always work very well [314]. Instead, several ideas emerged that proved to generally improve the training of the supervised task. Among the predilect solutions, transfer learning and pre-training using an unsupervised criterion are based on transferring some information from an unrelated task and/or dataset to help solve the current problem. Unsupervised pre-training of the deep neural networks resulted in improved performance over state-of-the-art methods for arrhythmia classification on benchmark datasets [137], [138], [321].

In this case study, similar to case study 1, the hypoglycemia detection problem was cast to a classification problem, in which the inputs were the individual, extracted heartbeats and the output was the corresponding glucose level event (low/normal). However, alternatively, a CDAE was employed for the unsupervised learning of robust, lower dimensional representation of the input signal (i.e. heartbeat). The results presented in this section showed that once a 'good' representation was learned, the extracted features were highly discriminative with respect to the glucose levels and that using the learned encoder parameters to initialise a CNN, yielded superior classification performance than just using a randomly initialised CNN. Therefore the contributions of this sections are as follows:
• Introduction of a novel approach for learning unsupervised ECG heartbeat features that were used in a clustering task. Moreover, this section shows that the learned features could be used for data visualisation in a lower dimensional space and the ECG features related to glucose levels were indeed the most significant ones.

• The performance of the CDAE was evaluated for the hypoglycemia detection task during the night, using data collected from 8 healthy, elderly participants (over 60) during a 36/24 hour stay in the HMRU.

• Different autoencoder embedding sizes were evaluated in a series of experiments, showing that 20D embeddings were enough to capture important heartbeat features.

• Tested and provided evidence for the hypothesis that employing a CDAE for the CNN parameters’ initialisation, the classification performance for hypoglycemia detection improved over the random initialisation of the CNN.

4.4.1 Experimental protocol

Twenty-five healthy, male participants over 60, who volunteered to participate in a research study investigating the impact of a 12-week exercise and nutrition intervention on healthy ageing, were continuously monitored during a 36 hour visit at the University Hospitals Coventry & Warwickshire in the HMRU. Due to safety reasons, the 36 hours spent in the HMRU was amended to 24 hours after \( \approx 5 \) months of running the study; therefore 10 volunteers spent only 24 hours in the HMRU, as illustrated in Figure 4.13. Out of the total 25 participants enrolled in the study, only the data corresponding to 8 participants could be used in this case study, due to multiple exclusion criteria, presented in Figure 4.13 including: noisy ECG, and lack of low glucose events during the monitoring nights. As in the previous case study, it was expected that a high number of participants were excluded from the analysis due to missing low glucose events, which is common in healthy individuals. However, the excluded participants do not bias the results as the models presented in this case study follow personalised medicine approaches. The participants underwent a baseline metabolic assessment and they were monitored for the next 36 or 24 hours in a calorimeter room (starting from 8 PM). Nominal 24h ECG was recorded with a wearable commercial device (Medtronic Zephyr BioPatch\textsuperscript{TM} HP) during the 36/24 hours in the calorimeter room, without removing the sensor. Participants also wore a CGM (i.e. FreeStyle Libre Flash) for measuring their interstitial glucose
levels every 15 minutes. CGM data corresponding to the first night of the experi-
ment (from 8 PM to midnight) were disregarded due to the reported decreased
accuracy in the first few hours of the CGM readings [268]. Moreover, to account for
the known circadian ECG changes during the night, the analysis was limited to the
nighttime (from midnight to 9 AM) period [307], [308].

4.4.1.1 Ethical approval

The study protocol was approved by the Ethics Committee of the University of
Coventry (reference: P59723), UK, and each person enrolled gave written informed
consent to participate. Moreover, the trial was registered on clinicaltrials.gov
(identifier number NCT03299972).
4.4.2 Dataset

ECG excerpts of 15 minutes were annotated as corresponding to normal or low glucose according to the CGM readings. Lower glucose level (i.e. LGL) episodes were defined as glucose concentration values lower than 3.9 mmol/L and a normal glucose level (i.e. NGL) was defined as a glucose concentration between 3.9 mmol/L and 7.5 mmol/L as per international guidelines [309]. As presented in Table 4.2, the cutoff limit for the hypoglycemia is slightly different in Europe and the US, however a value below 4 mmol/L is agreed upon. In this study the limit of 3.9 mmol/L was the predefined hypoglycemia limit that the CGM devices used, limit established by the clinicians involved in the study. Moreover, the strategy of excluding the heartbeats close to the hypoglycemia threshold during training accounted for the CGM reading error and for the variability in the hypoglycemia limit. As in case study 1, section 4.3.2, the first step was to isolate each heartbeat. This was achieved by detecting a fiducial point (i.e., the R peak) and then selecting a window of time of 640ms around the fiducial point. The R peak for each heartbeat was detected using a QRS detection algorithm as proposed in [246] and isolating 160 samples around the R peak (i.e. 60 samples preceding the R peak and 100 samples following the R peak). Due to the known CGM reading lag [268], each heartbeat was associated with the glucose value that corresponded to the current timestamp of the heartbeat + 5 minutes. Finally, 5-minute ECG excerpts of the same event (low/normal) corresponding to each participant were extracted and split randomly into 3 different datasets: training (50%), testing (the remaining 50%) and validation (25% out of training dataset). The 5-minute ECG segments that were associated a glucose value between the considered hypoglycemia threshold of 3.9 mmol/L and 4.2 mmol/L were ignored during training and were only considered during model evaluation. As in the previous case study, the proposed model is person-specific, so the model was trained from scratch and tested only on data corresponding to the same person.

4.4.3 DL architecture, experimental settings

A CDAE, as described in chapter 2, section 2.4, was employed in this study, the specific CDAE structure used for this case study is shown in Figure 4.14. The CDAE optimisation objective used was the squared error between the input heartbeat and the reconstructed one, therefore the accuracy of the fit was assessed by the mean squared error MSE. The convolutions performed preserved the size of the input, thus the size of the output feature maps obtained from either performing
a convolution or a deconvolution were the same, equal to the size of the input $x$. In this study, the performance of the autoencoder was evaluated by testing how useful the learned representations were for a certain task of interest, in this case for detecting low glucose events. Therefore, a good representation was defined as one that yielded a more accurate classifier.

Both the encoder and decoder were mapped as CNNs that consisted of 5 convolutional layers each comprising 30 filters. The first 2 convolutional layers had a size of 5 and the following 3 filters had size 3. The convolution was performed by preserving the input size, with stride 1. Experiments with convolutions without padding and with a stride of 2 were also performed, therefore reducing the input dimension to 33 for the last feature maps, however the performance of the classifier decreased. ReLU was used as activation function in the CDAE, except for a linear activation in the fully-connected layers.

The last feature maps of the encoder were flattened and connected to a fully-connected layer containing either 2, 20 or 50 neurons (to test different compression rates). The decoder was a symmetric CNN, having the exact same structure as the encoder but in reverse order and performing a deconvolution operation (transposed convolution).

The input of the encoder was a z-normalised raw heartbeat that was corrupted with a noise factor of 0.5 with Gaussian noise ($\mu = 0, \sigma = 1$). The classifier consisted of 3 fully-connected layers of sizes 300, 150 and 2 that were added on top of the encoder. The two output glucose levels (low or normal) to be detected by analysing the ECG were the outputs of the classifier. The learning rate for training the CDAE was set to $10^{-2}$ and $10^{-3}$ for the classifier and manually tuned using the validation dataset, the batch size for both feature extraction step and classification was 500. The used optimiser for both tasks was the AdamOptimizer [99], the CDAE parameters were initialised using the Xavier initialiser [311] whereas the fully-connected layers of the classifier were initialised randomly with values from a Gaussian distribution ($\mu = 0, \sigma = 0.05$).

The hyper-parameters of the classifier were chosen by performing a combination of grid search and manual tuning on the validation dataset for the following: number of CNN layers (tested 2 layers up to 8), number of filters in each layer (tested values between 10 and 50), filter sizes (tested values between 2 and 10 ), number of neurons in the FC layers. In order to choose the number of optimisation iterations for the CDAE several experiments were performed for one of the participants, that assessed the performance of the validation dataset for 100 ($\sim 1$ epoch, depending on the number of heartbeats extracted for each participant) up to 1000 iterations.
From this analysis, it resulted that 100 iterations were enough for the CDAE to learn and extract useful features, thus all the classification results were obtained after 100 CDAE training steps. The same training and testing datasets were used in both models: CDAE and for the low glucose classifier.

\textit{t-SNE} was configured with a perplexity of 25 and the number of optimisation iterations was set to 5000, following the known recommendations [322].

### 4.4.4 Classification

To build a low glucose detection classifier a stack of fully connected layers was added on the top of the encoder part of the CDAE, as illustrated in Figure 4.14. The weights and the biases learned during the training of the autoencoder were used as initial values for the CNN part of the classifier, which had the same structure as the encoder. The classifier outputs were given by a softmax activation and the loss to be optimised was the cross-entropy.
4.4.5 Unsupervised clustering using the learned embeddings

Once an autoencoder has been trained in an unsupervised fashion, its highest level representation could be used as input to a supervised algorithm such as an ANN, SVM, Logistic Regression, etc. Alternatively, in this study, the interest was to test whether the produced representations were relevant for low glucose detection task but without supervision, therefore, the t-SNE [92] method was employed on the reduced representations in a clustering task.

4.4.6 Results

4.4.6.1 CDAE

4.4.6.1.1 Reconstruction errors  Figure 4.15 presents an example of the CDAE output for a single randomly chosen heartbeat from the test dataset for two randomly chosen subjects (ID 6 and ID 7). The same heartbeat was reconstructed using different CDAE embedding sizes: 2D, 20D and 50D. The overall MSE for all the subjects and their corresponding heartbeats is presented in Figure 4.16. Therefore, the mean squared error (MSE) and mean absolute error (MAE) computed on the test heartbeats for 100, 500 and 1000 CDAE optimisation iterations for all the subjects are presented in Figure 4.16. Figure 4.15 shows that even with a low embedding dimension (2D) the heartbeats could be accurately reconstructed by the CDAE. In fact comparing between 20D and 50D embedding sizes, the reconstructed heartbeat overlapped almost perfectly on the input heartbeat, visually the difference could not be assessed.

4.4.6.2 Dimensionality reduction

A nonlinear dimension reduction method (t-SNE) was employed to visualise the data in lower dimensional space. Specifically, the automatically learned embeddings (y=20D and y=50D) obtained for a random subsample (max size = 10000) of the test heartbeats were used as inputs to t-SNE. Figure 4.17 presents the results of t-SNE applied to 20D (a) and 50D (b) embedding space for one of the subjects (ID = 6) and the 2-D scatter plot of the 2D learned embeddings (c). The colour associated with each point in Figure 4.17 corresponds to the glucose level (red for low (\(\leq 3.9\) mmol/L) and green for normal (\(> 3.9\) mmol/L) associated with each heartbeat. In order to differentiate the actual glucose levels, a gradient colouring was used, thus higher glucose levels are illustrated with a darker green whereas the lower glucose values with a darker red. Therefore, the t-SNE plot not only facilitates the visualisation of the input data in a lower dimensional space, but it also enables
Figure 4.15: Example of reconstructed heartbeats for different embedding sizes, for two of the participants

(d) ID = 7, emb = 2D  (e) ID = 7, emb = 20D  (f) ID = 7, emb = 50D

Figure 4.16: MSE and MAE for the output heartbeat reconstruction on the test dataset for all the study participants. The boxen plots show the MSE and MAE for different embedding sizes (2D, 20D and 50D), and for 100, 500, and 1000 CDAE optimisation iterations.
Figure 4.17: Visualisation of the learned embeddings for the participant with ID=6. For 20D and 50D the t-SNE method was employed to reduce the input dimensionality of the embedding to 3D. t-SNE was optimised for 5000 steps and the perplexity parameter was set to 25.

The quality assessment of the information preserved in the CDAE embeddings with respect to certain outcomes or output variables. The results presented in Figure 4.17 are thus remarkable, as clear heartbeats clusters by glucose levels can be visualised, although so far the CDAE was trained in an unsupervised manner without considering any labels.

4.4.6.3 Classification

A series of experiments was performed to test the influence of the number of CDAE iteration steps and the size of the embeddings on the classification performance of the heartbeats by glucose levels. Moreover, a CNN network was also tested for classification without the pre-training step, thus having all the parameters initialised randomly. Figure 4.18a and Figure 4.18b present the classification AUC on the
validation and testing datasets respectively for all the CDAE considered iterations (100, 500, 1000) and for all the considered embedding sizes (2D, 20D and 50D). The blue bar in the plot corresponds to the CNN trained end-to-end without the pre-training step.

According to the results presented in Figure 4.18, the number of the CDAE iterations did not have a significant influence on the classification results, therefore Tables 4.9 and 4.10 present detailed classification results (i.e. sensitivity, specificity, accuracy, auc) for 100 CDAE iterations and a 20D embedding size.

Finally, Figure 4.19 illustrates the predicted glucose levels during the night for two of the participants. Similar plots for all the study volunteers can be found in the annexe B, section B.2. The illustrated waveform represents the ground-truth glucose level, as recorded by the CGM. Each coloured point corresponds to a 5-minute prediction when employing the same majority voting strategy described in the previous case study where red points correspond to low glucose level prediction and green points to normal glucose level prediction. Lighter colour intensities correspond to less certain predictions (i.e. where the 5-minute mean prediction probability for a specific class $\leq 0.7$).
Table 4.9: Classification results on the test dataset using 2D, 20D and 50D embedding sizes in the CDAE and 100 optimisation steps of the CDAE. The reported values represent % / 100.

<table>
<thead>
<tr>
<th>ID</th>
<th>2D</th>
<th>20D</th>
<th>50D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sens</td>
<td>Spec</td>
<td>Acc</td>
</tr>
<tr>
<td>1</td>
<td>0.896</td>
<td>0.755</td>
<td>0.810</td>
</tr>
<tr>
<td>2</td>
<td>0.860</td>
<td>0.806</td>
<td>0.845</td>
</tr>
<tr>
<td>3</td>
<td>0.819</td>
<td>0.945</td>
<td>0.921</td>
</tr>
<tr>
<td>4</td>
<td>0.651</td>
<td>0.999</td>
<td>0.936</td>
</tr>
<tr>
<td>5</td>
<td>0.971</td>
<td>0.862</td>
<td>0.934</td>
</tr>
<tr>
<td>6</td>
<td>0.947</td>
<td>0.964</td>
<td>0.961</td>
</tr>
<tr>
<td>7</td>
<td>0.973</td>
<td>0.647</td>
<td>0.891</td>
</tr>
<tr>
<td>8</td>
<td>1.000</td>
<td>0.927</td>
<td>0.943</td>
</tr>
<tr>
<td>Average</td>
<td>0.890</td>
<td>0.983</td>
<td>0.905</td>
</tr>
</tbody>
</table>

Table 4.10: Classification results on the validation dataset using 2D, 20D and 50D embedding sizes in the CDAE and 100 optimisation steps of the CDAE. The reported values represent % / 100.

<table>
<thead>
<tr>
<th>ID</th>
<th>2D</th>
<th>20D</th>
<th>50D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sens</td>
<td>Spec</td>
<td>Acc</td>
</tr>
<tr>
<td>1</td>
<td>1.000</td>
<td>0.926</td>
<td>0.959</td>
</tr>
<tr>
<td>2</td>
<td>0.987</td>
<td>1.000</td>
<td>0.991</td>
</tr>
<tr>
<td>3</td>
<td>0.995</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>4</td>
<td>1.000</td>
<td>0.979</td>
<td>0.980</td>
</tr>
<tr>
<td>5</td>
<td>0.999</td>
<td>1.000</td>
<td>0.999</td>
</tr>
<tr>
<td>6</td>
<td>0.654</td>
<td>0.976</td>
<td>0.911</td>
</tr>
<tr>
<td>7</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>8</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>Average</td>
<td>0.954</td>
<td>0.985</td>
<td>0.980</td>
</tr>
</tbody>
</table>

4.4.7 Discussion

In this case study, a non-invasive, non-intrusive method for nocturnal low glucose level detection in healthy individuals over 60, based on ECG analysis recorded with wearable devices was proposed. In contrast to the work in the previous section, (i.e. 4.3), here an unsupervised method for feature extraction was investigated, testing the influence of the learned features on the accuracy of the classification task. Moreover, the results presented in section 4.4.6 show that the low dimensional embeddings learned by the CDAE, used in an unsupervised setting reveal clear heartbeat clusters, grouped by glucose levels. It was also investigated how different embedding sizes and CDAE optimisation steps could affect the supervised task and showed that initialising a CNN with the learned filters of a CDAE can boost the performance of the classifier, also confirmed by the experiments performed in this case study.
Figure 4.19: Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval. Red points correspond to predicted low glucose level whereas the green point represent the normal glucose predictions. Lighter color intensities (red or green) denote a lower probability of the prediction $\leq 0.7$

As emphasised before, due to high individual differences in the ECG and in the hypoglycemia ECG related changes [36], [299], person-specific ECG-based systems are appropriate for accurate hypoglycemia detection. To provide further evidence, Table 4.11 presents a summary of previous results obtained for hypoglycemia detection using different machine learning algorithms, based on manually extracted ECG features in comparison to the results obtained in this study. However, due to protocol differences: person-specific vs non-person specific, DL vs feature-based models, healthy vs type 1 diabetics, this study cannot be directly related to previous research, thus the results presented in Table 4.11 are roughly informative, but still revealing that the proposed method in this study led to promising results, potentially opening new research directions.
Table 4.11: Recent hypoglycemia detection studies based on ECG analysis results

<table>
<thead>
<tr>
<th>Study</th>
<th>Health condition</th>
<th>Method</th>
<th>Sens. (%)</th>
<th>Spec. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>[290]</td>
<td>Type 1</td>
<td>Extreme learning</td>
<td>78.00</td>
<td>60.00</td>
</tr>
<tr>
<td>[319]</td>
<td>Type 1</td>
<td>Deep Belief Network</td>
<td>79.70</td>
<td>50.00</td>
</tr>
<tr>
<td>[295]</td>
<td>Type 1</td>
<td>Particle swarm optimisation</td>
<td>63.64</td>
<td>54.67</td>
</tr>
<tr>
<td>[293]</td>
<td>Type 1</td>
<td>Second order multiple regression</td>
<td>75.00</td>
<td>50.23</td>
</tr>
<tr>
<td>[317]</td>
<td>Type 1</td>
<td>Fuzzy inference system</td>
<td>75.00</td>
<td>51.64</td>
</tr>
<tr>
<td>[318]</td>
<td>Type 1</td>
<td>Multiple regression-based</td>
<td>79.07</td>
<td>53.64</td>
</tr>
</tbody>
</table>

Proposed
avg Healthy CDAE + CNN 90.00 90.06

Figure 4.15 and Figure 4.16 present the CDAE reconstruction performance of the input heartbeats. The sample reconstructed heartbeats in Figure 4.15 show that even with a low-dimensional embedding space (2D), the CDAE was capable of a precise reconstruction of the inputs. As expected, increasing the latent space dimension to 20D, the reconstruction error decreased, as the CDAE aims to maximise the variance of the data in the latent space to achieve a low reconstruction error. The results presented in Figure 4.16 were obtained on the heartbeats included in the testing dataset, therefore, these heartbeats were not shown to the CDAE during the training phase. Interestingly, when the latent space dimension was further increased to 50D, on average, both the MSE and the MAE did not seem to improve in comparison to the 20D latent space. This shows that the CDAE might overfit the training data when using additional variables in the latent space.

To further investigate the learned embeddings, a dimensionality reduction technique (t-SNE) was employed to obtain a 3D visualisation of the learned embeddings (y) during the CDAE training, presented in Figure 4.17a and Figure 4.17b. As explained in chapter 2.4, t-SNE aims to retain the local structure of the data in the latent space, so it attempts to create a separation between the natural clusters in the data. However, so far no glucose level information was provided to either the CDAE or t-SNE, both CDAE and t-SNE were optimised in an unsupervised way. The results presented in Figure 4.17 are thus remarkable, as it can be observed that t-SNE revealed clear clusters of heartbeats that correspond to low and normal glucose levels, highlighted in Figure 4.17 by different colours (red for low glucose and green for a normal glucose level). Moreover, the colour gradients in Figure 4.17 corresponded to different levels of glucose for that specific class, where a light red corresponded to a higher glucose level of the low glucose class and a light green colour corresponded to lower normal glucose level. The heartbeats clusters presented in Figure 4.17a and Fig. 4.17b not only separated the low and normal glucose classes but revealed
that the normal glucose heartbeats could be further separated into different levels of normal glucose for the 20D embeddings. When employing t-SNE for 50D embeddings, the learned features evidently allow the visualisation of the heartbeats as a progression according to the glucose levels. These results are indeed exceptional as the CDAE was provided with neither classes or structure within the classes (i.e. the actual glucose level associated with each heartbeat). The 2D scatterplot of the learned embeddings can still reveal a separation between the higher and lower values of glucose in the normal class; however, less well separated than for the higher dimensional latent space. The motivation for this is that autoencoders in general, do not construct low-dimensional data representations that can widely separate the data into the natural classes, mainly because the an autoencoder aims to minimise the reconstruction error, thus maximising the variance of the data [93].

Table 4.9 and Table 4.10 present the classification results obtained for each participant when using a CNN classifier initialised with both random parameters and learned parameters from training the CDAE. No significant improvement was observed when optimising the CDAE for a larger number of steps (500 or 1000) as shown in Figure 4.18a, therefore the presented classification results were obtained using the 20D embedding size and 100 CDAE optimisation iterations. Both individual and average classification results were promising, the average sensitivity and specificity were 90% and 90.6% respectively on the testing dataset for the 20D embeddings network. Similar to the reconstruction MSE, using 50D embeddings did not significantly improve the classification results, the average sensitivity increased from 90% to 90.16%; however, specificity decreased from 90.06% to 89.9%. Figure 4.19 shows the 5-minutes predictions during the night for two randomly chosen subjects. These prediction plots are very useful as they reveal important information such as: the prediction confidence, what time certain glucose events occurred, when the misclassification 5-minute ECG occurred in relation to the measured glucose levels, etc., all these would be disregarded when just assessing the classification performance.

Fig. 4.18 illustrates the area under the curve (AUC) as a summary measure for the classification performance, for different CDAE iterations and different embedding sizes, for all the participants. Moreover, the blue bar illustrates the classification performance when the CNN classifier was randomly initialised, instead of using the CDAE learned parameters. Figure 4.18 shows that the AUC was on average lower for the CNN without pre-training on both training and testing datasets. Thus, in accordance with previous evidence [86], [89], [90], [137], [138], the low glucose detection task benefits from using a pre-trained CNN classifier.
4.4.8 Case study 2 - Conclusions

A novel method for nocturnal low glucose detection in healthy elderly subjects based on ECG signal has been presented. The proposed system was evaluated on 8 healthy participants during a 1 or 2 night stay at the University Hospitals Coventry & Warwickshire in the HMRU, using two wearable devices for recording a one lead ECG signal and continuous glucose levels. The low glucose levels were detected on average with 90% specificity and sensitivity for a 5-minutes prediction resolution. The unsupervised learned embeddings allowed the organisation of the testing heartbeats into distinctive clusters, even showing a progression along the actual glucose levels. These results, are remarkable as they show that the unsupervised learned features were discriminative for different glucose levels and that other factors contributing to heartbeat morphological variations (activity, breathing rate, etc.) were less intense in comparison to the ECG alterations caused by hypoglycemia.

However, the study has several limitations that should be investigated in future research. Firstly, only a relatively small number of healthy participants were used to validate the results. Secondly, for developing a real-life hypoglycemia detection system, more data are required, also collected in free-living conditions. Being inside in an HMRU, that is $3m^2$ room, definitely affected the participants’ behaviour and limited their physical activity. Moreover, this study could be performed as similar ECG alterations were observed in both healthy and diabetics; however, low glucose levels are scarce in healthy, and the ultimate beneficiaries of a hypoglycemia alarming system are diabetic patients. Therefore, the proposed system should also be tested on type 1 and type 2 diabetics in order to validate further the current results.
4.5 Conclusions

This chapter presented a novel, non-invasive method for nocturnal hypoglycemia detection in healthy using wearable devices. Two different experiments were performed, the first one carried out in free-living conditions whereas the second experiment was carried out at the University Hospitals Coventry & Warwickshire in the Human Metabolism Research Unit (HMRU). In total ECG data from 12 participants has been analysed, providing evidence for the feasibility of the development of a real-time hypoglycemia alarming system. Different DL methodologies were tested, to account for different hypotheses: the morphology of an individual heartbeat and the sequence of heartbeats in a short ECG excerpt are informative about the low glucose levels. Moreover, the second case study introduced an additional, unsupervised step in training the CNN based model that showed potential improvements in the heartbeat classification task. Also, the unsupervised step was driven by the scarcity of data in the second case study, where only 24 or 36 hours of ECG recordings were available.

The content of this chapter is the subject of a patent application, submitted on the 2\textsuperscript{nd} of September 2019 to the UK Intellectual Property Office, application No. 1912487.4. Moreover, two publications discussing the results of the two case studies presented in this chapter have been submitted to Scientific Reports and Biomedical Signal Processing and Control journals in September 2019. The publication submitted to Scientific Reports has been accepted for publication and available online [9].
Chapter 5

Deep Learning for Congestive Heart Failure Diagnosis using ECG Recordings

5.1 Chapter overview

In this chapter, the time series classification framework that was proposed earlier, in chapter 4, is employed for a different case study: diagnosis of CHF. The application of the same framework with minor architectural changes for a different problem than the one presented in chapter 4, shows the generality of the proposed method, that can be applied to cardiovascular data, investigating different conditions. The common characteristics shared by the low blood glucose detection problem and the CHF detection are:

- the classification setup
- the inputs which represent constant length time series, in particular, the inputs for both case studies are the individual heartbeats extracted from a long-term ECG recording, as both CHF and low glucose levels have a signature on the heartbeat morphology
- the output targets which are a discrete set of classes

Thus, diagnosis of CHF using DL methods is chosen as a case study not only because it is a serious condition with high prevalence in developed countries, but it also represents an illustrative example of a highly researched problem in the machine learning community, that has focused over the years on finding the most informative
features for discriminating between healthy persons and CHF patients, that can benefit from the representation learning capability of the DL methods.

The chapter presents a brief review of the methods in the literature for the diagnosis of CHF. In addition to subject classification, the proposed method allows the evaluation of each heartbeat for a specific subject. The visualisation techniques introduced in section 2.6 were also applied in this study, highlighting the sequences in the input heartbeat that led to a specific classification.

5.2 Background on Heart Failure (HF) assessment

CHF is a pathophysiological condition responsible for the failure of the heart in pumping blood in the body [323] which has encountered widespread research and societal attention [324]. According to the European Society of Cardiology, around 26 million people worldwide are affected by a form of heart failure [325]. CHF is a strongly degenerative condition, and its prevalence increases quickly with age [326], [327]. The mortality rate is closely associated with the degree of severity, reaching peaks of 40% in the most serious events [e.g., New York Heart Association (NYHA) classes III-IV] [328]. CHF is also one of the foremost reasons for hospitalisation in the elderly, and it is characterised by a resilient relapse rate, with half of the outpatients readmitted within a few months from hospital discharge [329]. Moreover, just among the most industrialised countries, the healthcare expenditure for CHF consumes 2-3% of the healthcare budgets, with the cost of hospitalisation being the greatest proportion of the spending [330]–[332]. Thus, with a worldwide ageing population and sustained pressures on healthcare systems and resources, there is the compelling demand—among patients, healthcare providers, policymakers, and society as a whole—to address this scenario by identifying highly accurate methods to improve the detection of heart failures [333] and in turn enable early and more efficient diagnoses. A comprehensive review of the tasks and methods related to HF management is presented in [333] and illustrated in Figure 5.1.

Recently, research has made significant progress in these areas. In particular, given the quantity and complexity of data involved, machine learning techniques and classifiers (e.g., SVM, MLP, k-Nearest Neighbours (k-NN), CART, Random Forest) [334]–[340] have been successfully applied to analyse, detect, and classify heart failures, as discussed and showed in previous works [334], [335], [341]–[343]. These approaches are able to distinguish between heart failure and healthy subjects are mostly based on HRV—the variation over time of the period between consecutive heartbeats extracted from ECG signals [344], showing that depressed HRV
patterns represent accurate markers for detecting the condition. However, building accurate HRV-based models is time-consuming and prone to error steps, due to the preprocessing and the iterative process of manually selecting appropriate features. Moreover, the best performing HRV-based models generally require either long-term signals (i.e., 24h) or at least the combination of short-term HRV with non-standard long-term HRV features, as shown in [345].

To tackle these issues, a novel framework of CHF detection is presented that does not rely on CHF features, rather it only uses raw ECG signals. This method is based on a 1-D CNN. As discussed in section 2.4, CNNs are hierarchical neural networks that mimic the human visual system and have proven to be effective in recognising patterns and structures of input data in image classification, localisation, and detection tasks, among others [22], [60], [76]. Moreover, they have been extensively used for time series analysis in classification tasks. Some successful examples include, among others: general time series classification [70], [102], [346], speech recognition tasks [63], arrhythmia detection [157], [301], and multivariate diagnostic measurements modelling [347], [348].

Inspired by this growing body of research, the study aimed to detect CHF through a 1-D CNN approach on the ECG signals. Adding to the significant benefit of building upon raw physiological data, this method enables visualisation of the input time series subsequences that are class discriminative (i.e., CHF vs healthy subjects). This feature considerably improves the interpretability of the CNN model and represents a crucial aspect of ensuring the ‘transparency’ of the method [349], [350].

Figure 5.1: Overview of the heart failure assessment studies, similar to [333]
Such transparency is fundamental to help researchers explain how conclusions are reached, and to aid professionals in better understanding correlations of pathophysiological behaviours and properties revealed by the model.

A form of class activation mapping (Grad-CAM) [79] was employed for highlighting the class discriminative regions in the input data. In other words, Grad-CAM uses the model’s last activation maps to originate a heat map that can be overlapped with the input, thus showing what regions in the input data contribute most to CHF detection. Overall, the proposed framework puts forward several developments for CHF detection, such as refraining from using hefty preprocessing and feature selection steps of HRV-based models, as well as enabling visualisation of the subsequences in the input time series which are used to reach certain clinically-relevant conclusions.

Inspired by the recent success of DL methodologies this chapter presents an accurate method based on the raw ECG signal for CHF detection, using a CNN based model.

5.3 Dataset

The study in this chapter represents a retrospective analysis on two publicly available datasets. The data for the normal subjects were retrieved from the MIT-BIH Normal Sinus Rhythm Database [1] included in PhysioNet [3]. This dataset includes 18 long-term ECG recordings of normal, healthy not-arrhythmic subjects (Females =13; age range: 20 to 50). The data for the CHF group were retrieved from the BIDMC Congestive Heart Failure Database [2] from PhysioNet [3]. This dataset includes long-term ECG recordings of 15 subjects with severe CHF (i.e., NYHA classes III-IV) (Females =4; age range: 22 to 63).

Altogether, the pool of data available for this study consists of about 20 hours of ECG recordings per subject; it contains two-channel ECG signals sampled at 250 samples per second for the BIDMC dataset, and 128 samples per second for the MIT-BIH dataset. The two datasets used in this study were published by the same laboratory, the Beth Israel Deaconess Medical Center, and were digitised using the same procedures according to the signal specification line in the header file. These datasets have been widely used in several studies concerning CHF detection, as explained in [333]. Indeed, it is an acknowledged practice to build and validate CHF classification models on these sources to ensure opportunities for reproducibility.
Table 5.1: Total number of extracted heartbeats from both datasets: MIT-BIH and BIDMC

<table>
<thead>
<tr>
<th>Normal heartbeats (18 subjects)</th>
<th>CHF heartbeats (15 subjects)</th>
</tr>
</thead>
<tbody>
<tr>
<td>275,974</td>
<td>214,531</td>
</tr>
</tbody>
</table>

5.4 Data pre-processing

The ECG recordings from the BIDMC dataset were down-sampled in order to match the sampling frequency of the ECG signals from the MIT-BIH dataset (i.e., 128 Hz). Records from both databases were already made available with computed beat annotations, which were used to isolate and extract individual heartbeats. A window of 235 ms before the annotated R peak (equivalent to 30 samples=128 samples/s*0.235 s), and a window of 390 ms after the R peak (equivalent to 50 samples) were considered to isolate each heartbeat. Only the heartbeats that were annotated as normal (N) were retained for further analysis. Each heartbeat was z-normalised to obtain an approximately zero mean value (i.e., amplitude) and a standard deviation close to 1. Because the number of heartbeats extracted for each subject was very large (70,000 heartbeats), and because temporally close heartbeats hold similar patterns, one single heartbeat was randomly selected every 5 seconds of the ECG signal in order to speed up the training process, without biasing the results.. The total number of heartbeats used in this work is shown in table 5.1. The number of non-normal heartbeats is very low in comparison to the normal ones, on average the non-normal heartbeats are below 1% of the total number of heartbeats available. A snapshot of normal and CHF heartbeats are presented in Figure 5.9. Figure 5.9 presents an overview of the average shape of the normal and CHF heartbeats included in the testing dataset.

5.5 Heartbeat classification

Each heartbeat was labelled with a binary value of 1 or 0 (hereafter 'class' not to be confused with the NYHA classes) according to the status of the subject: healthy or suffering from CHF, respectively. As customary in ML, the dataset was randomly split into three smaller subsets for training, validation, and testing (corresponding to 50%, 25%, and 25% of the total data, respectively). Because one person’s heart-
beats are highly intercorrelated, these were included only in one of the subsets (i.e., training, testing, or validation set) at a time. In order to reduce the variance of the classification results, the random splitting process was repeated ten times. In this way, the CNN was trained and evaluated over ten runs; the reported results represent the average performance on these runs, unless otherwise stated. Performing ten different random splits of the dataset, and then averaging the results of the models, is a similar procedure to that of cross-validation (i.e., averaging the variance and bias of the estimator). The sole difference between the two methods is that the additional validation dataset is used for early stopping of the training process and tuning the hyperparameters of the CNN. Two different performance evaluation strategies were employed: a) individual beat classification, and b) classification on 5 minutes of ECG excerpts through what was called a 'majority voting scheme'. Specifically, by denoting the number of individual heartbeats classified as normal with $N_1$, and the number of heartbeats that were classified as CHF with $N_0$ within 5-minute ECG excerpts, the final class associated with a given 5-minutes ECG excerpt was calculated as $\max(N_0, N_1)$.
5.5.1 Performance measures

The following measures were computed for binary classification to assess the performance of the CNN classifier: Accuracy (Acc), Precision (P), Sensitivity (Sens), Specificity (Spec) and Area under the Curve (AUC) [345]. These performance indicators correspond to the gold standard in the literature [17], [339] thereby enabling ongoing model comparison.

5.6 Proposed CNN architecture

The complete architecture of the proposed model is shown in Figure 5.3. A 1-D CNN classifier was used for feature extraction, and the MLP module for the classification of the raw ECG heartbeats. For the beat classification problem, the CNN acts as a feature extractor block. The final activations obtained from the last layer of convolution are used as inputs in a MLP network. The final output is obtained from a softmax layer [73]. The basic convolutional layer is followed by a batch normalisation layer [107] and a ReLU activation function [312]. The convolution operation is performed in each layer by 20 1-D filters (i.e., kernels) with sizes 10, 15, and 20 using a stride of 1, hyperparameters, as in previous studies [70], [301]. Thus, the convolutional block can be formalised as a set of three operations—convolution, batch normalisation, and non-linear activation—defined by the equations below:

\[
\begin{align*}
    y &= W \ast x + b \\
    bn &= BN(y) \\
    act &= ReLU(bn)
\end{align*}
\]

(5.1)

where \( \ast \) is the convolution operator, \( W \) are the convolutional layer parameters (i.e., the learnable filters), \( x \) represents the input time series, \( b \) the bias, \( BN \) the batch normalisation function, and \( ReLU \) the activation function as defined in section 2.6. The resulting CNN is built by stacking three convolutional blocks, each comprising 20 filters of sizes 10, 15, and 20.

The final activations are flattened and passed to an MLP with a hidden layer of 30 neurons that are then fully connected to the output neurons. The final label is obtained from the softmax function. The pooling operation is excluded from the convolutional blocks according to recent research showing that it does not affect the classifier performance and might affect overfitting (see, e.g., ResNet [351]).
5.6.1 Visualisation of class specific sequences

The Grad-CAM technique was employed to visualise the subsequences in the input ECG heartbeats to discern a certain class \([79]\). A detailed description of Grad-CAM method is provided in section 2.4. Briefly, for Grad-CAM the weights used in the linear combination of the forward activation maps are computed as the global average pooling of the gradients of the score for class \(c\) with respect to the last feature maps. Specifically, the method implies the computation of the gradient of the score for class \(c\) \((y_c)\) with respect to the feature maps \(A\) (i.e., in our case the feature maps of the last convolutional layer) that are global-average-pooled to obtain the weights \(\alpha_k^c\). Therefore, the weights \(\alpha_k^c\) are computed as follows:

\[
\alpha_k^c = \frac{1}{m} \sum_i \sum_j \frac{\partial y_c}{\partial A_{ij}^k}
\]

(5.2)

where the \(\alpha_k^c\) capture the importance of the feature map \(k\) for a target class \(c\). The Grad-CAM heatmap for a class \(c\), is obtained as a weighted combination of feature maps, followed by ReLU:

\[
GradCAM_{m}ap_c = ReLU \left( \sum_k \alpha_k^c A_k \right).
\]

(5.3)

As shown in Figure 5.3, Grad-CAM methods were used to derive the class activation...
map for class \( c \), which indicates the importance of the activation at a temporal location \( x_i \); this leads to the classification of the input time series as a class \( c \).

### 5.6.2 Experimental setting

As presented in chapter 3, the state of the art methods for pattern recognition in high dimensional data such as time series are based on DL methods. Therefore, this chapter leverages a DL-based model, in particular a CNN for the automatic ECG features learning to diagnose CHF. As previously mentioned the CNN network consists of 3 convolutional blocks, each comprising 20 filters of sizes 10, 15 and 20. The size of the hidden layer of the MLP network is 30 neurons. The optimisation function used was the AdamOptimizer [99] with an initial learning rate of 1e-3. The batch size was set to 200. The maximum number of training steps was set to 3000, which represent approximately 2.5 epochs, considering the batch size of 200 and the approximate training dataset size of 250k heartbeats. The validation dataset was used to monitor the training process and to account for the occurrence of the overfitting. In such a case, an early stopping criterion was implemented, as presented in section 2.6. This criterion stopped the training if the AUC for the validation dataset did not improve during a number of k=30 optimisation step. The loss function used was the categorical cross-entropy. The weights were initialised using the Xavier initialiser [105], and the biases were initialised to 0. The network was implemented in Python using the TensorFlow [310] framework.

Due to the high number of hyperparameters (the number of CNN layers, number of filters in a CNN layer and their size, the learning rate, batch size, number of fully-connected layers) in the CNN, a combination of architecture and hyperparameters was assessed through an iterative process, using a grid-search and manual tuning. As regards the architecture structure, this was searched over the number of convolutional layers (3 to 9), different filter sizes (from 5 to 20), and the number of filters in each convolutional layer (5 to maximum 30). The learning rate was instead manually tuned to achieve faster convergence; the considered values were \( 10^1 \) to \( 10^5 \). The results presented below were obtained with the CNN architecture that achieved the highest performance on the validation dataset with the minimum number of parameters.
5.7 Results

5.7.1 Individual heartbeat classification

Table 5.2 reports the performance for individual beat classification as the average of the 10 different runs for the 10 random splits of the input data for training, validation, and testing.

Table 5.2: Average per-heartbeat classification results as a mean of 10 runs on 10 random splits of the input data into training/validation/testing datasets. Results presented as % / 100.

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Validation</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Accuracy</strong></td>
<td>$0.999 \pm 1.1 \times 10^{-16}$</td>
<td>$0.982 \pm 2.0 \times 10^{-3}$</td>
<td>$0.978 \pm 2.0 \times 10^{-3}$</td>
</tr>
<tr>
<td><strong>AUC</strong></td>
<td>$0.999 \pm 1.1 \times 10^{-16}$</td>
<td>$0.990 \pm 1.1 \times 10^{-16}$</td>
<td>$0.980 \pm 1.1 \times 10^{-16}$</td>
</tr>
<tr>
<td><strong>Sensitivity</strong></td>
<td>$0.999 \pm 1.1 \times 10^{-16}$</td>
<td>$0.973 \pm 2.0 \times 10^{-3}$</td>
<td>$0.963 \pm 4.0 \times 10^{-3}$</td>
</tr>
<tr>
<td><strong>Specificity</strong></td>
<td>$0.999 \pm 1.1 \times 10^{-16}$</td>
<td>$0.978 \pm 2.0 \times 10^{-3}$</td>
<td>$0.986 \pm 7.0 \times 10^{-4}$</td>
</tr>
<tr>
<td><strong>Precision</strong></td>
<td>$0.999 \pm 1.1 \times 10^{-16}$</td>
<td>$0.970 \pm 3.0 \times 10^{-3}$</td>
<td>$0.985 \pm 7.0 \times 10^{-4}$</td>
</tr>
</tbody>
</table>

Table 5.3 shows a confusion matrix for the mean of the ten runs, without considering the voting strategy, corresponding to the testing dataset. These findings show that on average the number of false positives is around 1% of the mean number of normal heartbeats, while the false negatives are around 3% of the total number of heartbeats in CHF class.

Table 5.3: Mean confusion matrix for the 10 different runs

<table>
<thead>
<tr>
<th></th>
<th>True CHF</th>
<th>True normal</th>
<th>Total Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classified CHF</td>
<td>67181</td>
<td>694</td>
<td></td>
</tr>
<tr>
<td>Classified normal</td>
<td>2071</td>
<td>61512</td>
<td></td>
</tr>
<tr>
<td>Classified correctly from each class</td>
<td>≈ 97.0%</td>
<td>≈ 98.8%</td>
<td>≈ 97.9%</td>
</tr>
<tr>
<td>Error</td>
<td>≈ 3%</td>
<td>≈ 1.1%</td>
<td>≈ 2.1%</td>
</tr>
</tbody>
</table>

A closer analysis for one of the ten models revealed that out of the total 854 misclassified CHF heartbeats, 614 belonged to the same subject (id:chf01). The same applied to the control group, where 478 beats that were misclassified belonged to the same subject (id:16272). Nonetheless, the number of misclassified heartbeats
for these two subjects was negligible, representing around 2%-4% of the total number of heartbeats that were extracted for the two subjects. In other words, subjects 16272 and chf01 could still be correctly categorised as healthy and CHF subjects, respectively because more than 95% of their heartbeats were appropriately classified. The 5 consecutive heartbeats were accepted to be very similar, thus averaging them should not result in significant changes in the heartbeat morphology. An additional simulation was performed to confirm this hypothesis in which the heartbeats over 5 seconds interval were averaged. As expected, no change in the performance of the classifier was observed.

5.7.2 Classification results using the majority voting strategy

The 'majority voting scheme' was employed for the classification of the heartbeats within a 5-minute ECG timeframe to enable direct comparison with previous studies using short-term HRV features computed on 5-minute ECG segments (see Section 5.5 for details). In short, the class associated with a 5-minutes ECG segment was the majority class of the individual heartbeats in that segment. Using such a voting strategy, the accuracy of the proposed method could be evaluated in discriminating between healthy and CHF patients when using 5-minute ECG recordings. All the performance measures considered (i.e., ACC, Sens, Spec, and AUC) improved to 99% when employing the 5-minute voting in comparison to the individual heartbeat predictions. The better performance of the 5-minute voting scheme is explained by the isolated misclassified single heartbeats, therefore averaging the predictions over a 5-minute ECG interval addresses the problem of sparse misclassified heartbeats. Moreover, the mean number of the misclassified 5-minutes ECG excerpts using the voting scheme on the ten runs was 23 out of a total of 2,227, which corresponds to about 0.1%. These results suggest that, different from existing evidence [345], a highly accurate model for CHF diagnosis can be built by using relatively short (5-minute) raw ECG recordings. To further corroborate this argument, the subjects and specific timeframes where any misclassified 5-minute ECG segment occurred were also investigated, as shown in Table 5.4. As a matter of illustration, here it is presented a randomly selected run. It can be observed that out of the total number of 16 subjects included in the test and validation datasets, only 5 of them present misclassified 5-minute ECG segments. 

Figure 5.4 presents a closer inspection of these results. It illustrates the heartbeat classification for one of the ten models used in the training, validation, and testing datasets including all the subjects within this study. Once again, it can be readily seen that the extracted heartbeats were correctly classified (i.e., the misclas-
Table 5.4: Test subjects with misclassified 5-minute ECG segments

<table>
<thead>
<tr>
<th>Subject ID</th>
<th>ECG segment start time → end time</th>
<th>Subject ID</th>
<th>ECG segment start time → end time</th>
</tr>
</thead>
<tbody>
<tr>
<td>16795</td>
<td>13:30 → 13:40</td>
<td>19093</td>
<td>4:35 → 4:40</td>
</tr>
<tr>
<td>16795</td>
<td>14:30 → 14:40</td>
<td>19830</td>
<td>10:40 → 10:45</td>
</tr>
<tr>
<td>16795</td>
<td>15:30 → 15:40</td>
<td>16272</td>
<td>16:05 → 16:10</td>
</tr>
<tr>
<td>16795</td>
<td>15:50 → 15:55</td>
<td>16272</td>
<td>16:15 → 17:05</td>
</tr>
<tr>
<td>16795</td>
<td>16:05 → 16:55</td>
<td>16272</td>
<td>17:30 → 17:35</td>
</tr>
</tbody>
</table>

Misclassified heartbeats for some subjects are considerably less than the correctly classified ones. Moreover, the misclassified beats show a consecutive trend, which is most likely indicating a noisy ECG segment in the raw recording. It might also be possible that these specific heartbeats were intrinsically not-discriminative, an issue that could be tackled by extending the training dataset. The misclassified ECG excerpts illustrated in Figure 5.5 indicates that the model failed to classify either noisy segments (e.g., subjects 16272, 19830) or ECG excerpts affected by movement artifacts, as also acknowledged by independent cardiological assessments. This circumstance can easily be confirmed by inspecting the data with the online viewer LightWAVE [352] from Physionet.

5.7.3 DL model motivation

According to new clinical studies [353]–[355], major ECG changes are expected in patients that were diagnosed with CHF, therefore the chosen DL architecture was based on a CNN. In this study, the actual sequence of heartbeats was considered to be less important than the heartbeat shape itself, in comparison to the previous studies that focused on quantifying the discrimination power of HRV features for diagnosing CHF. Moreover, the CNN architecture employed in this study is simpler than the one presented in chapter 4, the proposed CNN had fewer number of convolutional layers, fewer number of filters and no weights regularisation was used. These choices are motivated by the excellent results obtained, in other words the model performance showed that the model’s capacity did not need to be increased. Figure 5.6 presents the loss value during training iterations, which suggests that the learning rate was chosen appropriately, as the loss decreases over time, asymptotic
Figure 5.4: Heartbeats classification for the subjects included in the training, validation and test datasets, evaluated on one of the 10 runs. Panels b) and d) present the predictions for the voting in a 5-minute ECG window. The red and green bars represent the extracted heartbeats in chronological order for CHF patients and healthy subjects, respectively. The grey gaps in the middle of the bars represent missing heartbeats.

to 0, without getting stuck at a certain value after the initial decrease.

The gradients and activation histograms that correspond to the last convolutional layer are presented in Figure 5.7 and Figure 5.8 respectively, show that the neurons are not completely saturated at 0. This ensures that the weights were correctly initialised, and the batch normalisation is preventing the neurons from being saturated.
Figure 5.5: Example of misclassified ECG segments evaluated on the validation and testing datasets

Figure 5.6: Loss during training

5.7.4 Class-specific sequences

Regarding the visualisation feature of the model, Figure 5.9 represents a summary of key regions in the input beats, obtained through Grad-CAM, that are class discriminative. The two ECG heartbeats in Figure 5.9 represent the average heartbeats included in the testing dataset corresponding to the normal (i.e. green) and CHF (i.e. red) classes with their corresponding error bands. The histograms represent the data points in the input ECG heartbeats above 0.8 in the normalised heat maps obtained through Grad-CAM. The sample points that were found to be significant (i.e., where the histogram points reached a threshold of 0.25) are presented in Figure 5.9. In other words, these points contributed the most to identifying a certain class.
Figure 5.7: Histograms of the gradients of the last convolutional layer, for each training step. It can be observed that the neurons are not completely saturated at 0, and that the gradients are preserving a normal distribution with zero mean.

Figure 5.8: Histograms of the activations in the last convolutional layer, for each training step. This plot shows that both the Xavier weight initialisation and the batch normalisation are preventing the neurons from being saturated.
Figure 5.9: Histograms obtained from the Grad-CAM heat maps, indicating the most important sample points in the input beats for a certain class. Each bar in the histograms represents a single sample point in the heartbeat time series, thus the greater the bar the more important that point is for the classification result.

5.8 Discussion

This study presents a novel and innovative framework for CHF detection based on CNN. While CHF detection using machine learning on ECG has traditionally relied on the extraction of HRV features to build classifiers able to discriminate between healthy and CHF subjects, in this study the body of knowledge was advanced by using as input data raw ECG heartbeats. The development of such a model allows one to bypass HRV features extraction and selection steps, yet, and importantly, without compromising the classification performance of the model. Indeed, as a matter of comparison, the model proposed by Isler et al. [336] based on short-term HRV features extracted from a randomly chosen 5-minutes ECG excerpt out of the total 24h ECG recordings, reports a classification performance of 96% sensitivity and specificity. Thus, not only has the classification performance been improved, but the results of this study showed that 99% of the 5-minute ECG excerpts could be used for providing a correct CHF diagnosis. Therefore, these benefits could be due to the ability of CNNs to automatically extract and learn patterns in the input data, rather than relying on specialised features that might not fully capture the true characteristics of the underlying physiological signals. In this study, a
shallow architecture (i.e., with 3 convolutional layers) was purposely chosen, yet using larger filter sizes. Experiments performed with different CNN architectures revealed that similar classification results could also be obtained with a deeper architecture and smaller filter sizes, also shown in chapter 3, section 3.3. For this case study, the number of neurons in the MLP layer did have great influence on the model’s performance and a much faster convergence was observed with batch normalisation than without. Another relevant note concerns the difference between this work and that by Acharya et al. [152] who employed a CNN-based model for CHF detection. The current study puts forward important benefits. First, as regards the evaluation methodology, the authors in [152] used the recordings corresponding to a certain subject for both training and testing the model; here, instead, only the extracted heartbeats in either training or testing datasets were considered for a run. Therefore, the validation methodology proposed in this study is more restrictive and it reflects the real life conditions, where data are not available beforehand for new patients that were not included in the training dataset. Considering an intra-subject evaluation usually results in over-optimistic results that do not reflect the generalisation performance of the model. That is, the subjects were divided into training and testing subsets, rather than using their corresponding ECG heartbeats. Moreover, different from the architecture proposed in [152], the model proposed in this study is simpler because it contains 3 convolutional layers only and no pooling operations, thus reducing the computational complexity to the benefit of possible future mobile applications. The CNN inputs are different too: in the current study, individual heartbeats were extracted, whereas in [152], ECG excerpts of 2 seconds were used as inputs. This represents a critical strategy that the proposed model advances because it allows one to investigate the heartbeat morphology in connection to the CHF diagnosis through the Grad-CAM method (see Figure 5.9). Therefore, to the best of my knowledge, this is the first time that discriminative segments in the input heartbeat are revealed for the CHF diagnosis. Finally, mean differences in the heartbeat morphology are clearly shown in Figure 5.9 in which the histograms highlight the regions in the input heartbeats that were mostly used by the CNN in making the predictions. Thus, the model proposed here, allows visualising those features that are automatically learned during the classification process. According to a recent study by Taylor and Hobbs [356], heart failure is associated with abnormal ECG in 89% of the cases. Moreover, James et al. [357] showed that heart failure patients have significantly prolonged QRS duration, prolonged QT duration, prolonged QTc and more rightward T wave axis. Also, de Luna et al. [358] reported long QT interval and visible variations of the
shape, amplitude or polarity of the T wave on an alternate-beat basis as ECG changes associated with heart failure. In Figure 5.9, two heartbeats were presented, obtained as averages of all normal and CHF heartbeats included in the available datasets. It can be appreciated that some of the morphological changes reported in the literature can be easily identified in Figure 5.9. These include, among others, a flattened, prolonged T wave and visible changes in the amplitude of the T wave. The highlighted parts in the normal and CHF heartbeats represent the ECG heartbeat segments used by the CNN network to generate a prediction. Whilst not entering into clinical interpretations, it can be observed that the identified important parts in the ECG are mainly located near the fiducial points (the Q, R and T waves).

As shown in Table 5.5, the classification results suggest that this method is highly competitive with, and in some respect superior to, previous models using HRV features. It is worth noticing that, while the classification process presented here is performed at the heartbeat level, the results obtained can be easily compared with previous studies by performing a per-subject classification, using the presented voting scheme, either for 5-minute ECG excerpts or on the entire ECG recording (i.e., by quantifying the majority class predicted for all the heartbeats corresponding to each subject).

In the proposed model, the error in the subjects classification is nil: Each subject is correctly classified as being either healthy or affected by CHF. Previously, Liu et al. [337] presented what to the best of my knowledge is the only other system achieving a similar accuracy in CHF detection. Yet, this work focuses on a purposely selected sample of subjects rather than on the entire pool available in the original datasets. The misclassified heartbeats are trivial in comparison to the correctly classified ones, indicating 100% accuracy for the classification at the subject level, as evident in Figure 5.4.

Added to their superior performance, the results presented here bring forward the intuition that short ECG recordings, of just about 5 minutes, can be sufficient to correctly diagnose severe CHF. This is an important result because, with the increasing availability of wearable devices capturing interim ECG recordings (e.g., smart-watches), accurate CHF detection and prediction might be soon performed through devices people carry everyday. These devices have a small physiological impact as they already play an important role in our daily life. In this respect, current approaches have looked at HRV, which is conventionally analysed using 5-minute excerpts, if not longer (e.g., nominally 24h). However, physiological phenomena running over much shorter timespans might not be fully captured when considering such long excerpts. Thus, by using individually extracted heartbeats to provide an
Table 5.5: Classification performance of the classifier proposed in the current study and those from other relevant previous studies

<table>
<thead>
<tr>
<th>Author</th>
<th>Method</th>
<th>Data</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asyali et al. [339]</td>
<td>Linear Discriminant Analysis, Bayesian Classifier Based on HRV features</td>
<td>1. Normal sinus rhythm RR interval database 2. Congestive heart failure RR interval database</td>
<td>Sensitivity: 0.82 Specificity: 0.98 Accuracy: 0.93</td>
</tr>
<tr>
<td>Isler et al. [336]</td>
<td>k-NN based on HRV features</td>
<td>1. Normal sinus rhythm RR interval database 2. Congestive heart failure RR interval database</td>
<td>Sensitivity: 0.96 Specificity: 0.96 Accuracy: 0.96</td>
</tr>
<tr>
<td>Mellilo et al. [334]</td>
<td>Classification and Regression Trees (CART) based on HRV features</td>
<td>1. Normal sinus rhythm RR interval database 2. Congestive heart failure RR interval database 3. Congestive heart failure RR interval database 4. BIDMB congestive heart failure</td>
<td>Sensitivity: 0.9 Specificity: 1.0 Accuracy: 0.96</td>
</tr>
<tr>
<td>Acharya et al. [152]</td>
<td>Raw, 2 seconds ECG CNN based method</td>
<td>1. MIT-BIH normal sinus rhythm database 2. BIDMB congestive heart failure 3. Fantasia</td>
<td>Sensitivity: 1.0 Specificity: 1.0 Accuracy: 1.0</td>
</tr>
<tr>
<td>Current study</td>
<td>Raw ECG heartbeats CNN based method</td>
<td>1. MIT-BIH normal sinus rhythm database 2. BIDMB congestive heart failure</td>
<td>Sensitivity: 1.0 Specificity: 1.0 Accuracy: 1.0</td>
</tr>
</tbody>
</table>

accurate diagnostic prediction, the approach proposed here facilitates future implementation of real-time systems towards the realisation of a Clinical Decision Support System (CDDS). In the CDDS envisioned end-users are not only experienced cardiologists, but also patients, caregivers, general practitioners, nurses, and so forth. Moreover, once trained, the network responds in a very short time and can be embedded into mobile phones or deployed to cloud architectures, as already discussed elsewhere [335], [359].

This study must also be seen in the light of its limitations, which anyway represent valuable calls for future research avenues. First, the CHF subjects used in this study suffer from severe CHF only (i.e., NYHA classes III-IV); the discrimination could yield less accurate results for milder CHF (i.e., NYHA classes I-II). Second, the use of acknowledged public datasets including raw ECG signals was aimed at improving the opportunity for comparison and transparency of science. Yet, this choice necessarily resulted in a dataset which, at the subject level, is on average smaller than those used in other models. Future studies will need to extend the re-
sults presented here in larger datasets. Finally, future research is encouraged as well as clinical practice to apply this framework to the pre-diagnostic assessment of CHF. This strategy will not only allow further model validation, but it will also enable moving from retrospective detection to prospective diagnosis, potentially improving human well-being, reducing healthcare costs, and hopefully even saving lives.

5.9 Conclusions

This study proposed a novel and highly-effective method for CHF detection based on CNNs. Compared to existing models, the added value of the proposed model is that it uses raw ECG signals, rather than HRV features, and yields prominent accuracy in detecting CHF. Indeed, it has been shown that the performance of the CNN model holds high scores when the classification is performed at the heartbeat level, and it is error-free as pertains the subject classification component. Moreover, this is the first study using advanced machine learning approaches to reveal which morphological characteristics of the ECG heartbeats are the most important to efficiently detect CHF. I believe that this might be an important contribution to clinical practice because clinicians, who are ultimately responsible for CHF patients, are generally refractory to the adoption of methods that are not fully transparent in showing how certain decisions were reached. The methods proposed in this study respond to the issue of model transparency. However, translating the technology into clinical practice requires a complex process, overlooked by a team of stakeholders, as recently pointed out by Wiens et al. [360]

Finally, the model provided here could be used as a highly-effective and generalisable classification method for other time series classification tasks in medicine and beyond (e.g., in neuro-behavioural research [361]). Therefore, there is a call for future work to expand the results presented in this study towards improving human wellbeing and reducing current healthcare financial and societal burdens.
Chapter 6

Deep Learning for HRV
Features Estimation from Raw ECG Recordings

6.1 Chapter overview

This chapter presents a framework and tools developed to answer Research Question 4 as defined in chapter 1, section 1.3. Specifically, in this chapter, a time series regression framework is proposed, for the estimation of HRV features from the raw ECG signal. In comparison to the studies presented in chapters 4 and 5, where the elements of the target sequences could take values from a discrete set of values (i.e. classes), this case study presents a solution for a time series regression problem, for which the output targets are real values. Moreover, the CNN-based systems presented so far, in chapters 4 and 5, focused on treating the inputs as static data. The choice of these methods was driven by the problems to be solved (i.e. automatically extracting and learning patterns in the input time series). However, the problem with this approach is that the importance of time is not fully valued. By ignoring the time component and applying static models to time series data, important structures in the input data might be lost. The reason is that the context of the input time series is lost; the only time dependency captured by the model is within the input dimension. Therefore, in order to be able to capture longer-term dependencies either the input dimension should be increased or other DL modules should be employed, such as RNNs, that are specialised modules for learning sequential data (refer to section 2.4, RNNs), which also showed slightly improved results over the CNN-based model for the hypoglycemia classification problem in chapter 4.3.
This chapter presents a case study in which both the time and space components of the input time series should be captured. The results of the study are important from two different perspectives: it might open new research directions in the biomedical field, especially for applications where HRV is considered as a predictive variable and it also provides a general framework for modelling time and space dependencies for sequential input data.

6.2 Background on HRV computation

HRV describes the variations between consecutive heartbeats within a specific time-frame. HRV analysis attempts to assess cardiac autonomic regulation by quantifying the sinus rhythm variability [362], which is usually derived from the consecutive QRS intervals (RR) of the ECG (Figure 6.1). The rhythm of the heart is controlled by the sinoatrial (SA) node, which is modulated by both the sympathetic and parasympathetic branches of the autonomic nervous system. Sympathetic activity tends to increase heart rate (HR↑), and its response is slow (a few seconds) [363]. Parasympathetic activity, tends to decrease heart rate (HR↓) and mediates faster (between 0.2 and 0.6 seconds) [363]. Thus, the continuous modulation of the sympathetic and parasympathetic innervations results in variations in heart rate. The aim of HRV analysis is to examine the sinus rhythm modulated by the autonomic nervous system. Thus, a first step in deriving the HRV is the quantification of the heartbeat, which is usually evaluated as the time difference between consecutive QRS complexes. The most important algorithms for heartbeat detection were previously presented in chapter 3, section (3.3.5.4).
Once the QRS complexes have been estimated, the HRV series can be derived. The RR series (inter-heartbeat intervals) is obtained as differences between successive R peaks occurrence times, as shown in Figure 6.1. The RR intervals are sometimes called NN intervals to indicate the interval between normal-to-normal heartbeats. A flow chart summarising the steps used for processing the ECG signal in order to obtain the HRV is presented in Figure 6.2.

Based on the RR series, four different categories of methods are usually applied for HRV analysis: time-domain, frequency-domain, nonlinear and time-varying methods. Time-domain measures treat the normal sinus to normal sinus (NN) interval as an unordered set of intervals and employ different statistical methods to express the variance of such data. The frequency-domain measures perform a power spectral analysis of the ordered NN intervals and show how these NN intervals are distributed as a function of frequency. The most common time and frequency-domain HRV features computed are presented in Tables 6.1 and Table 6.2 respectively. The nonlinear characteristics of HRV are usually analysed using measures such as: a Poincaré plot, approximate and sample entropy, detrended fluctuation analysis, correlation dimension, recurrence plots and wavelets [248]. The time-varying methods refer to methods that quantify the trends for certain time-domain and frequency-domain measures.

Over the recent decades, HRV analysis has become a popular method for the assessment of the autonomous nervous system in diverse fields of research. Depressed HRV has been proven to be an independent predictor for several clinical outcomes, such as death in chronic congestive heart failure [39], myocardial infarction, mental stress detection [40], risk of accidental falls in hypertensive patients [41], early warning sign of diabetic neuropathy [42], and many others.
spectral density (PSD) analysis provides the basic information of how power (i.e., variance) distributes as a function of frequency. It is possible to calculate the power spectrum of absolute differences between adjacent NN intervals. The number of pairs of adjacent NN intervals differing by more than 50 ms in the entire recording can be used to locate a stable and noise-independent baseline.

Table 6.2: Common frequency domain HRV measures, taken from [364]

<table>
<thead>
<tr>
<th>Variable</th>
<th>Units</th>
<th>Description</th>
<th>Frequency range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total power</td>
<td>ms²</td>
<td>The variance of NN intervals over the temporal segment</td>
<td>approximately ≤ 0.4 Hz</td>
</tr>
<tr>
<td>VLF</td>
<td>ms²</td>
<td>Power in very low frequency range</td>
<td>≤ 0.04 Hz</td>
</tr>
<tr>
<td>LF norm</td>
<td>n.u.</td>
<td>LF power in normalised units</td>
<td>0.04–0.15 Hz</td>
</tr>
<tr>
<td>HF norm</td>
<td>n.u.</td>
<td>HF power in normalised units</td>
<td>0.15–0.4 Hz</td>
</tr>
<tr>
<td>LF/HF norm</td>
<td>Ratio</td>
<td>LF [ms²]/HF [ms²]</td>
<td>0.5–2.0</td>
</tr>
</tbody>
</table>

Table 6.1: Common time domain HRV measures, taken from [364]

<table>
<thead>
<tr>
<th>Variable</th>
<th>Units</th>
<th>Description</th>
<th>Frequency range</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 min total</td>
<td>ms²</td>
<td>The variance of NN intervals over the temporal segment</td>
<td>approximately ≤ 0.4 Hz</td>
</tr>
<tr>
<td>VLF</td>
<td>ms²</td>
<td>Power in very low frequency range</td>
<td>≤ 0.04 Hz</td>
</tr>
<tr>
<td>LF norm</td>
<td>n.u.</td>
<td>LF power in normalised units</td>
<td>0.04–0.15 Hz</td>
</tr>
<tr>
<td>HF norm</td>
<td>n.u.</td>
<td>HF power in normalised units</td>
<td>0.15–0.4 Hz</td>
</tr>
<tr>
<td>LF/HF norm</td>
<td>Ratio</td>
<td>LF [ms²]/HF [ms²]</td>
<td>0.5–2.0</td>
</tr>
</tbody>
</table>

A geometric measure, heart rate variability (HRV), quantifies the variability of the intervals between successive R peaks. The central tendency of the distribution is calculated by computing the mean and standard deviation of all NN intervals. The square root of the mean of the sum of the squares of NN intervals is the standard deviation of all NN intervals (SDNN).

### Geometric measures

- **TINN**: Baseline width of the minimum square difference triangular interpolation of the highest peak of the histogram of all NN intervals (details in Fig. 2.)
- **LF/HF Ratio**: HF [ms²]/LF [ms²]
- **HF norm**: HF power in normalised units
- **HF/(Total Power–VLF)**: HF power in normalised units
- **LF/(Total Power–VLF)**: LF power in normalised units
- **LF norm**: LF power in normalised units
- **VLF norm**: VLF power in normalised units
- **ULF**: Power in the ultra low frequency range
- **VLF**: Power in the very low frequency range
- **LF**: Power in the low frequency range
- **HF**: Power in the high frequency range
- **α**: Slope of the linear interpolation of the spectrum in a log-log scale

### Statistical measures

- **SDNN**: Standard deviation of all NN intervals.
- **SDANN**: Standard deviation of the averages of NN intervals in all 5 min segments of the entire recording.
- **RMSSD**: The square root of the mean of the sum of the squares of differences between adjacent NN intervals.
- **SDNN index**: Standard deviation of all NN intervals in all 5 min segments of the entire recording.
- **NN 50 count**: Number of pairs of adjacent NN intervals differing by more than 50 ms in the entire recording.
- **pNN50**: NN 50 count divided by the total number of all NN intervals.

### Frequency domain measures

- **5 min total power**: The variance of NN intervals over the temporal segment.
- **VLF**: Power in very low frequency range.
- **LF**: Power in low frequency range.
- **HF**: Power in high frequency range.
- **ULF**: Power in the ultra low frequency range.
- **VLF**: Power in the very low frequency range.
- **LF**: Power in the low frequency range.
- **HF**: Power in the high frequency range.
- **α**: Slope of the linear interpolation of the spectrum in a log-log scale.

### Analysis of entire 24 h

- **Total power**: Variance of all NN intervals.
- **ULF**: Power in the ultra low frequency range.
- **VLF**: Power in the very low frequency range.
- **LF**: Power in the low frequency range.
- **HF**: Power in the high frequency range.

### Analysis of short-term recordings (5 min)

- **5 min total power**: The variance of NN intervals over the temporal segment.
- **VLF**: Power in very low frequency range.
- **LF**: Power in low frequency range.
- **HF**: Power in high frequency range.
- **ULF**: Power in the ultra low frequency range.
- **VLF**: Power in the very low frequency range.
- **LF**: Power in the low frequency range.
- **HF**: Power in the high frequency range.
- **α**: Slope of the linear interpolation of the spectrum in a log-log scale.
Commonly, both time-domain and frequency-domain features are computed using long-term (24-hour) ECG recordings. However, recent studies have shown that some of these features can be reliably computed and used from shorter ECG recordings (less than 5-minutes). The HRV features computed from an ECG segment that are less than 15 and 5-minutes are often referred to as short-term and ultra-short HRV features, respectively [39]. The need for reducing the ECG monitoring period is crucial, especially for real-time applications where decisions are usually taken from the analysis of the most recent ECG heartbeats. Shorter ECG recordings can be easily recorded without a significant increase in healthcare costs using wearable devices, and they can be easily translated in out-patient clinical life. Thus, being able to reduce the ECG recording interval without compromising the analysis results represents an important step towards using HRV analysis in real-time applications. This study aimed to evaluate the association between the time and space representations of the heartbeats and some of the commonly computed HRV features, to further reduce the number of heartbeats required for HRV estimation. Recent studies [16], [41] performed both short and ultra-short-term HRV analysis for the comparison of their prediction capability on different clinical outcomes.

Alternatively, the focus of this study was to investigate whether the HRV features computed for a 5-minutes ECG excerpt can be inferred using a shorter ECG interval. In order to achieve this, a framework based on a RNN was proposed, presented further in this chapter.

6.3 Dataset

This study was carried out using the publicly available dataset from the Sleep Heart Health Study (SHHS) [4]–[7]. The SHHS is a multi-centre cohort study implemented by the National Heart, Lung and Blood Institute (United States of America) to determine the cardiovascular problems of sleep-disordered breathing. Out of the total 6441 subjects enrolled in the study, a subset of 500 subjects with high-quality ECG recordings (125-Hz sampling frequency) was selected for a sub-study to quantify the HRV by sleep stage. QRS complexes (R-points) were detected using Compumedics (Abbotsford, VIC, Australia) Somte software Version 2.10 (Builds 99 to 101). The R-points were classified as normal sinus, supraventricular premature complex or ventricular premature complex. The automated annotations were reviewed by a trained technician who made appropriate corrections [4].

The computed 5-minute HRV features for the 500 subjects together with the original ECG recordings were made publicly available. It has been previously shown
that some of the HRV features are highly correlated with each other, thus for short-
term data only certain time-domain and frequency-domain measures can be reliably
computed [362]. Therefore, the computed HRV features for the sleep stage analysis,
that were made public are summarised in Table 6.3. Considering the features corre-
lation observation, this study focused only on the prediction of four HRV features:
AVNN, SDNN, HF and IHR.

The available data for each subject comprise of an average of 7 hours of ECG
recordings, which represents on average around 85 intervals of consecutive 5-minute
ECG segments together with the corresponding HRV features for each segment. A
histogram presenting the distribution of the 5-minute segments for the 500 subjects
is presented in Figure 6.3. The zero heartbeats represent gaps in the recorded data,
therefore the restriction of the RR interval between 0.75 and 1.5 seconds ensured
that the analysis was carried out on normal resting heart rates.

### 6.4 Data pre-processing

The ECG recordings were segmented into consecutive 5-minute excerpts with no
overlap, followed by a QRS-complex detection algorithm similar to the one proposed
in [246]. In order to filter out the noisy segments from the actual heartbeats, an
additional check was performed for each extracted heartbeat. Specifically, it was
checked that the minimum and the maximum values of each detected heartbeat lie
close to the annotated R peak. In case this condition was violated, the heartbeat
was discarded from the list of heartbeats that correspond to the 5-minute interval.
In addition to restricting the analysis to RR intervals < 1.5 seconds and > 0.75
seconds, only the 5-minute ECG excerpts with at least 200 heartbeats, but no more
than 400, were analysed. The distribution of the number of heartbeats in the 5-
minute intervals is presented in Figure 6.4. The final number of filtered 5-minute
Table 6.4: Total number of 5-minutes intervals considered in training, validation and testing datasets

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Validation</th>
<th>Testing</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>21945</td>
<td>2718</td>
<td>2282</td>
<td>29945</td>
</tr>
</tbody>
</table>

intervals used in the study was 29945, that was split into three different datasets for training (21945), validation (2718) and testing (2282) of the proposed model, shown in Table 6.4. The number of the 5-minute ECG intervals for each dataset was obtained by randomly splitting the subjects into these three different groups (400 subjects for training, 100 for validation and testing) then performing the above filtering steps, also illustrated in figure 6.5. Thus, the proposed system was trained and tested on different subjects.

Once the individual heartbeats were isolated, each heartbeat was filtered with a low-pass 3\textsuperscript{rd} order Butterworth filter (after performing experiments using different filter orders and observed that the 3\textsuperscript{rd} order filter preserves the shape and the amplitudes of the heartbeat waves) and afterwards standardised using: \( x(t) = \frac{x(t) - \mu}{\sigma} \). A summary of the preprocessing steps is presented in Figure 6.5.

6.5 Proposed DL architecture

For the HRV features inference problem, the input was sequential consisting of consecutive heartbeats in a 5-minute ECG excerpt. Thus, this study aimed to

Figure 6.3: Distribution of the 5-minute segments availability for the 500 subjects
investigate whether the LSTM-based networks are able to learn a sequence of ECG heartbeats for inferring the ultra-short-term HRV features. The intuition behind employing an LSTM-based model is that the time-varying heartbeat dynamics are explained by the HRV features and, as previously presented (refer to chapter 2, section 2.4), RNNs are specialised DL modules for learning temporal dependencies. The specific LSTM model employed in this study is shown in Figure 6.6.

The proposed framework consists of three parts: 1) data preprocessing, 2) aggregation of heartbeats over time, sequence-learning, using RNNs and 3) single/multiple stacked linear layers for performing linear regression. A different number of heartbeats was extracted from each segment and these were considered as inputs to the LSTM cells; the tested values were: 250, 200, 100 and 50. The values were chosen
in order to test the minimum number of heartbeats in 5-minute ECG required to infer the HRV features. The number of units in the LSTM cell was set to 60, the dense layer sizes were set to 100.

6.5.1 Architecture choice motivation

DL methods have been successfully employed for different time series analysis tasks, for problems such as classification and time series forecasting, detailed in chapter 3. RNNs in particular, due to the recurrent connections, are capable of large scale learning as shown in recent studies, being used for speech recognition [63], language translation models [62], or mental stress classification based on electroencephalogram signals, and have thus proved to be successful in learning temporal dependencies between the inputs. However, a significant limitation of the simple RNN models (discussed in more detail in section 2.4), which integrate state information over time, is known as the vanishing or exploding gradient effect, both referring to the ability of RNNs to backpropagate an error signal through a long-range temporal interval. The RNN version, known as LSTM, firstly proposed in [83], includes recurrent modules which enable long-range learning. LSTM units consist of hidden states augmented with nonlinear mechanisms to allow a state to propagate without modification, be updated, or be reset, using simple learned gating functions. Thus, an LSTM-based model was proposed in this study for the task of HRV features inference.
6.5.2 Experimental setting

For training the network, the mean squared error (MSE) was used as optimisation objective and model evaluation criterion. The learning rate was set to $10^{-1}$ and the used optimiser was Adam [99], the state of the art optimiser, as described in chapter 2. The learning rate was chosen after several experiments in which multiple values were tested over the range of $10^{-1}$ to $10^{-5}$. The batch size was set to 100, for a faster optimisation. Furthermore, dropout regularisation was used in all LSTM cells, with a probability of 0.5 for considering that cell. An early stopping criterion was employed for reducing overfitting, based on the minimum MSE obtained on the validation dataset. The model was trained for a maximum number of 15000 steps that corresponded to about 68 epochs. The regression performance was assessed using multiple measures: the MSE, median absolute error (MedAE), mean absolute error (MAE), the Pearson correlation coefficient (R) between the predictions and the correct HRV values and for measuring the agreement the Bland-Altman plot [365] was employed.

6.6 Results

The regression results obtained by employing the proposed LSTM architecture, using a maximum number of 250 time steps, for the selected four HRV features are presented in Table 6.5. The only two variables that could be predicted with a satisfactory error were AVNN and IHR, as shown in Table 6.5, therefore Table 6.6 shows the regression results only for AVNN and IHR when using fewer time steps: 200, 100 and 50. The results were obtained on the test dataset, after training the model for the maximum number of epochs. The results obtained when using the early stopping criterion were very similar to the presented ones, therefore the early stopping did not improve the regression results in this case study.

Table 6.5: Test dataset LSTM regression results corresponding to four HRV features: AVNN, SDNN, HF and IHR for 250 time steps (thus, equivalent of 250 input heartbeats)

<table>
<thead>
<tr>
<th>Variable</th>
<th>MSE</th>
<th>MedAE</th>
<th>Correlation (R)</th>
<th>MAE/Mean value (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVNN</td>
<td>3951</td>
<td>37.58ms</td>
<td>0.84</td>
<td>48.27ms/953.42ms (5%)</td>
</tr>
<tr>
<td>SDNN</td>
<td>615.5</td>
<td>13ms</td>
<td>0.60</td>
<td>17.55ms/52.1ms (33.6%)</td>
</tr>
<tr>
<td>HF</td>
<td>163.3</td>
<td>311ms²</td>
<td>0.04</td>
<td>552.9ms²/593.5ms² (93.2%)</td>
</tr>
<tr>
<td>IHR</td>
<td>13.4</td>
<td>2.09bpm</td>
<td>0.89</td>
<td>2.71bpm/63.86bpm (4.2%)</td>
</tr>
</tbody>
</table>
The number of samples that were predicted correctly on the test dataset within an error range of 5% or 10% of the correct value are presented in Table 6.7.

6.7 Discussion

The results presented in section 6.6 indicate that the aggregation of heartbeats across time using an LSTM network is effective for the inference of certain HRV features. Specifically, out of the 4 tested variables (AVNN, IHR, SDNN, HF), promising results were obtained for AVNN and IHR, both representing time-domain HRV features. Further experiments, including increasing the capacity of the model and testing different DL architectures are required in order to show that the frequency-domain features can also be accurately inferred from the raw ECG signal. The proposed system showed promising results only for the HRV time-domain features.

The prediction errors for the AVNN were between 5-6% when using 250 down to 50 heartbeats for a 5-minute ECG segment, which for AVNN is less than the known

Table 6.6: Test dataset - LSTM results obtained for AVNN and IHR features considering different number of time steps

<table>
<thead>
<tr>
<th># beats</th>
<th>AVNN</th>
<th></th>
<th>IHR</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>4743</td>
<td>39.69</td>
<td>0.79</td>
<td>51.8 (5.4%)</td>
<td>15.4</td>
</tr>
<tr>
<td>100</td>
<td>4728</td>
<td>41.43</td>
<td>0.8</td>
<td>52.3 (5.48%)</td>
<td>22.61</td>
</tr>
<tr>
<td>50</td>
<td>5437</td>
<td>46.55</td>
<td>0.78</td>
<td>57.0 (5.9%)</td>
<td>21.4</td>
</tr>
</tbody>
</table>
Table 6.7: Number of correct predictions for AVNN and IHR using a 5% and 10% error threshold

<table>
<thead>
<tr>
<th>Variable</th>
<th># of correct predictions out of a total of 2282 included in the test dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>5% error</td>
</tr>
<tr>
<td>AVNN</td>
<td>1362 (59.7%)</td>
</tr>
<tr>
<td>IHR</td>
<td>1565 (68.6%)</td>
</tr>
</tbody>
</table>

Figure 6.8: Bland Altman plot for the AVNN (a) and IHR (b) obtained on the test dataset

measurement error (i.e. < 10%) [248]. This implies that the AVNN could be predicted with an average error of 5%. If this error is satisfactory for certain studies then the proposed method can be used to infer the AVNN value by using around 50 ECG heartbeats (< 1 minute ECG), in contrast to a 5-minute ECG segment.

The IHR was predicted with less than 5% mean error even when using only 50 time steps or input heartbeats. Figure 6.7 (a and b) presents the errors distribution between the computed AVNN and IHR and the predicted values. It can be observed that for IHR the variance of the errors distribution is smaller than for AVNN and the peaks of the distribution lie very close to 0, which represents a perfect prediction. Figure 6.8 (a and b) present another view on the agreement between the measurement and the prediction, revealing that the majority of the points lie within ±1.96 std of the mean difference for both of the variables.
6.8 Conclusions

A DL model, based on an LSTM was developed and evaluated for the inference of certain HRV features, based on the raw ECG signal. Therefore, the results of this study show that instead of computing different HRV features and using them as predictive variables for different clinical outcomes, the raw ECG signal could be used instead for the prediction of the same conditions. The number of heartbeats that was necessary to predict some of the time domain features with a satisfactory MAE was investigated and showed that 50 input heartbeats were appropriate for two of the HRV variables prediction within 10% error. However, changing the network architecture and cascading a CNN with the LSTM are interesting directions to be explored in the future, in addition to testing other HRV features.
Chapter 7

Conclusions and Future Work

7.1 Chapter overview

This chapter presents the main conclusions of this thesis. Section 7.2 reiterates the scope, aims and objectives of this research. Section 7.3 restates the research gaps and questions that drove the work presented in this thesis. Moreover, section 7.3 provides a summary of the work and results obtained in order to address the research questions. Section 7.4 provides several recommendations for future work, based on the limitations and opportunities identified.

7.2 Research aim and objectives

Recent progress in the acquisition of digitised data, in the computing infrastructure, in the development of new ML algorithms has created unique opportunities for the development of AI applications, that promise to revolutionise many aspects of society, including medical practice. Prominent AI applications in healthcare were mainly focused on the interpretation of medical images, such as retinal fundus images for diagnosing age-related macular degeneration and diabetic retinopathy [25], or skin images to classify lesions as benign lesions or malignant skin cancers [26]. Both applications have been trained on hundreds of thousands of medically labelled images and validated against medical experts, showing that on average, the AI algorithms could outperform the domain experts. The US Food and Drug Administration has already approved the first ML application, Arterys Cardio DL, that leverages cloud computing and DL in a clinical setting to automatically segment the heart ventricles in cardiac magnetic resonance imaging.

Moreover, the 25th volume of Nature Medicine (2019) published two studies (i.e.
for automatic arrhythmia detection [157] and screening for cardiac contractile dysfunction [155]) and one review paper [28] demonstrating the success of ML in cardiology, specifically for the analysis of the ECG signal. Despite the fact that ECG is widely used in clinical practice for diagnosing multiple cardiac diseases, the visual inspection of ECG in real-time relies on certain discrete clinical features that cannot always objectively capture the diversity of ECG abnormalities and morphologies, as emphasised by Lyon et al. in [30]. Therefore, less subjective computational methods are required for detecting changes in the ECG that might be confusing for the human eye.

ECG analysis experienced a great increase in interest in the past few years being used in diverse applications (see chapter 3), mainly due to relatively cheap, non-invasive ECG acquisition, but also driven by sensor development advances. The research presented in this thesis focused on particular AI computational methods, namely DL techniques for developing several health monitoring applications. The ECG recordings used in the case studies presented in this thesis were either from benchmark datasets such as Physionet [3] or privately acquired with wearable devices. This research aimed to advance the knowledge and methods related to ECG analysis using DL approaches, with an emphasis on 2 different conditions: hypoglycemia detection in free-living conditions and CHF diagnosis.

Accordingly, the main objectives of this research were:

**Objective 1**: To review the main ECG signal-based applications that employed DL methods, identifying the optimal DL architectures, ECG preprocessing steps, and model validation strategies for specific tasks.

**Objective 2**: To develop a proof-of-concept for a personalised, non-invasive, ECG-based low glucose detection system using wearable devices, in free-living conditions using the representational power of DL techniques.

**Objective 3**: To investigate whether using an unsupervised step for pretraining the DL model improves the performance of the low blood glucose detection system.

**Objective 4**: To investigate the performance of DL models for CHF diagnosis on a benchmark dataset and to compare its performance with traditional feature-based models.

**Objective 5**: To determine whether HRV features can be inferred from a short (less than 1 minute), raw ECG excerpt by leveraging DL techniques.
Objective 6: To devise and validate a general method for visualisation of class-discriminative subsequences in the input time series, thus to enhance the model transparency when employing specific DL modules.

7.3 Research questions and answers

The objectives stated above derived from the set of research questions, identified through a literature review, presented in chapter 3. This section presents a summary of the work performed in this thesis in order to answer the research questions and to fill the identified research gaps, together with a synthesis of the main findings.

Research Question 1: How appropriate are DL-based methodologies for physiological data analysis, in particular for ECG analysis:
- is it feasible to use DL methods for automatic time series representation learning, for dimensionality reduction and extraction of relevant information?
- is it feasible to apply DL methods for physiological time series classification and regression?

As previously mentioned, in the past few years, the interest in ECG analysis applications that leveraged different AI approaches has been on the rise, as also revealed in recent reviews [11], [28]–[31]. The advances in digitally recorded ECG and the development of wearable sensors capable of accurately recording ECG signals, facilitated an easier, cheaper ECG acquisition. Traditional clinical ECG recorded on paper print-outs required manual digitisation; therefore, the development of computational methods for ECG analysis was not facilitated.

A great rise in interest for biomedical research, particularly for biomedical signal processing was encouraged by the publication of PhysioNet Resources [3] that offered free access to a large collection of physiological and clinical data and even to open-source software. Moreover, the PhysioBank archive that comprises of digitised recordings of physiological signals and time series including collections of cardiovascular, neural and other biomedical signals from both healthy individuals and patients with a variety of conditions: CHF, sleep apnoea, sudden cardiac death and many other contributed to a growing interest in ECG analysis, as demonstrated in chapter 3. Furthermore, the publication of the UCR multi-domain time series database [55] encouraged the development of computational methods for the TSC task.

159
Therefore, ECG data availability, together with the increased interest in AI-based methodologies for time series processing, resulted in an unprecedented diversity in ECG applications and approaches, published in the past few years.

Chapter 3 presented an original review performed in order to synthesize and to identify influential research areas and publications regarding the predilect ECG application domains, the recurrent DL-based models and their specific structure, the ECG pre-processing steps, the ECG segmentation methods. Therefore, a dataset including information regarding specific details about the employed methods was extracted from 106 recent studies on ECG analysis based on DL methods was used for the analysis. Specifically, in order to identify recurrent applications and the related analysis approaches; therefore, to provide an answer to the research question, two standard methods were performed: quantitative analysis and bibliometric analysis.

The results of the analysis above revealed that DL models have been successfully employed for diverse applications, including: ECG classification, myocardial infarction detection, QRS annotation, biometrics and may others as presented in chapter 3, Figure 3.10. Moreover, certain preferred DL models were identified, CNN-based networks being the most employed analysis method. RNN-based methods and AEs have been employed for applications such as ECG reconstruction, ECG compression, ECG noise reduction or detection of sleep-disordered breathing. In total more than thirty ECG-based applications that were carried out using thirty-three different public datasets and used more than fifteen DL methods for the ECG analysis were identified, illustrated in Figure 3.6 and Table 3.2. As a result of this diversity, the review also revealed a high heterogeneity across the structure of the DL models; therefore, the need for standardising certain hyperparameters was identified.

Research Question 2: By leveraging DL methods, is it feasible to develop a personalised, non-invasive, non-intrusive low glucose detection system based on raw ECG signals analysis, recorded with wearable devices? Would an unsupervised feature learning step improve the performance of the low glucose detection system?

As stated in chapter 3, the ECG recordings capture the propagation of the electrical signal in the heart from the body surface. Therefore, many cardiac structural and electrophysiological malformations determine changes in the ECG morphology and their identification and correct interpretation help clinicians to diagnose cardiac diseases. Moreover, the general behaviour of the cardiac contraction can also be altered by the variation of the physical and chemical composition of the fluids that surround the heart. Recently, clinical studies investigated the ECG alterations during hypoglycemia in both diabetics and healthy subjects, showing that low glucose
levels have a signature on ECG morphology [35], [36], [292], [308].

Hypoglycemia is known to affect the quality of life of diabetics, especially nocturnal hypoglycemia that causes significant anxiety, can cause nocturnal seizures and although rare, can even cause death during sleep [36].

Chapter 4 presented a study to investigate the feasibility of developing a personalised, non-invasive nocturnal hypoglycemia detection system by leveraging the ECG signal recorded with wearable devices and the representational power of DL methods. Two experiments were performed on different cohorts of healthy individuals, presented in chapter 4 in study 2 (section 4.3) and study 3 (section 4.4). Study 2 was carried out on a sample of 8 healthy individuals, in free-living conditions, recording continuously ECG signals and baseline glucose levels using wearable sensors during 14 consecutive days. Study 3 was carried out on a cohort of 25 healthy, elderly individuals, in a calorimeter room at the University Hospitals Coventry & Warwickshire, being monitored for either 24 or 36 hours. Therefore, study 3 was carried out in a more controlled environment, as participants’ movement was limited by the size of the calorimeter room (i.e. 3m²) and the meals were predefined. Similar DL methods, based on personalised medicine approach, were employed in both studies, tackling the inter-subject variability in the ECG and the individual alterations induced by the low glucose levels.

In study 2, different DL methodologies were tested (i.e. a CNN and a CNN + RNN model), to account for two different hypothesis: 1. the morphology of an individual heartbeat and 2. the sequence of heartbeats in a short ECG excerpt are informative about the low glucose levels. The results presented in study 2 showed that both the heartbeat morphology (Average accuracy on testing days: 81.8%) and the sequence of heartbeats (Average accuracy on testing days: 84.3%) could be used effectively for low glucose detection during the night, in everyday-living conditions. Therefore, the results of study 2 suggested that ECG recordings using wearable sensors could be used for the development of a low glucose detection system, showing great performance improvements over previous attempts that leveraged manually extracted ECG features.

Data collected for study 3 were used to provide further evidence supporting the feasibility of a non-invasive, ECG-based system for low glucose detection and to investigate certain unsupervised methods for improving the performance of the DL models. The unsupervised step consisted of an CDAE, trained to reconstruct the original, distorted input heartbeats. The learned parameters of the encoder part of the CDAE were used to initialise a CNN trained to classify the heartbeats by the glucose levels. The low glucose levels could be detected on average with 90% specificity.
and sensitivity for a 5-minute prediction resolution. Moreover, the unsupervised learned embeddings allowed the organisation of the heartbeats into distinctive clusters according to the glucose levels. Therefore the results presented in study 3, showed that the unsupervised learned features were discriminative for different glucose levels and that other potential factors contributing to heartbeat morphological variations (activity, breathing rate) were less discriminative in comparison to the ECG alterations caused by hypoglycemia.

**Research Question 3:** Using DL based methods, is it feasible to analyse the raw ECG for the diagnosis of CHF? Are DL based methods more accurate than the traditional feature-based methods?

CHF is a severe condition responsible for preventing the heart pumping blood in the body [323] which is associated with significant mortality and healthcare expenditure. [324]. The high prevalence, of over 23 million subjects worldwide and the frequent hospitalisations for CHF drove public interest towards building effective disease management strategies. Such policies require the analysis of large amounts of data for CHF prevention, early detection or severity assessment, that can improve the quality of life and reduce the associated healthcare costs [333].

Towards this direction, chapter 5 presented a study to investigate and develop a CHF diagnostic system based on raw ECG signals and DL models. The study presented in chapter 5 was carried out on a public dataset comprising of long-term ambulatory ECG recordings corresponding to both healthy subjects and individuals diagnosed with severe CHF (33 subjects in total, $\approx$ 20 hours of ECG/subject). The proposed DL method was based on a CNN, as CHF is known to alter the ECG heartbeat morphology.

The results of this study, showed a superior performance of the proposed system over the traditional feature-based methods, the subject-based classification accuracy was 100%. Added to their superior performance, the results presented in chapter 5 brought forward the intuition that short-term ECG recordings, of just about 5 minutes, could be sufficient to correctly diagnose severe CHF which could facilitate the implementation of a real-time CHF detection system. Moreover, this study employed the Grad-CAM method to reveal which morphological characteristics of the ECG heartbeats were the most important to efficiently detect CHF. Therefore, the implementation of the proposed method provides more transparency for the model development, which is important for the medical professionals who are in general resistant to the adoption of methods that are not fully transparent and interpretable.
Research Question 4: Using DL methods is it possible to infer certain 5-minute HRV features from a raw, shorter ECG excerpt?

Over recent decades HRV analysis has become a diffuse, non-invasive method for the assessment of autonomous nervous system imbalances. Therefore, low HRV has been associated with increased risk of anxiety and depression [43], a higher risk of accidental falls in hypertensive patients [41], early warning sign of diabetic neuropathy [42] and many other conditions.

Chapter 6 investigated whether several HRV features (AVNN, IHR, SDNN and HF) computed for a 5-minute ECG excerpt could be inferred using a shorter ECG interval. A public dataset comprising of 500 subjects with high quality ECG recordings [4]–[7] (≈ 7 hours of ECG/subject) was used in this study. The proposed method for the HRV features inference was based on a RNN, in which the input consisted of a variable number of consecutive heartbeats (i.e. 200, 100 or 50) and the output represented the HRV feature to be estimated. Therefore, in general terms the HRV estimation problem was cast as a regression task.

The results of this study showed that AVNN and IHR could be estimated with less than 6% error when using only 50 consecutive heartbeats from a 5-minute interval, which is less than the known 10% HRV computation error. Therefore, these results showed that a fewer number of heartbeats (≈ 1 minute) than the usual 5-minute ECG signal are required for the estimation of certain HRV features. Moreover, the results of the study advanced the idea that the computation of specific HRV features might be bypassed and real-time ECG applications could be facilitated due to shorter ECG segments used for the analysis.

Research Question 5: Is it possible to improve transparency and interpretability of DL models for time series analysis? In particular, is it possible to determine the essential subsequences in the ECG heartbeat that are used by the DL models for reaching specific outcomes?

Representation learning approaches, in particular DL models have proved to be successful in many application domains, including biomedical, as previously discussed in chapter 1 and chapter 3. Although these models could achieve performances that outranked the domain experts, it is not straightforward to explain the conclusions of the models, to debug the models (i.e. to identify weaknesses) or to interpret from a medical/biological perspective the model’s predictions. As a result, most often DL models are referred to as 'black boxes'. Moreover, careful attention should be aimed towards processing the input data in terms of data selection,
randomisation, training and testing datasets split in order to obtain an unbiased model. Recording physiological data in free-living conditions with wearable sensors results in large, noisy datasets that require filtering and data selection, operations that should be performed without considering the outcome to be predicted.

Therefore, in order to build trust in AI based systems and integrate them in our everyday lives, it is required to understand better how the DL models reached certain conclusions, in other words it is required to build AI systems that are ‘transparent’ and interpretable. While building transparent models might not be a problem for certain domains (e.g. object recognition in images), in the medical field model interpretability is crucial, in order for medical professionals to gain confidence in the model’s predictions. Therefore, in healthcare not only is the the performance of the models important but also the ability to explain the results obtained by these models in order to make them more understandable [11]. However, there is typically a trade-off between accuracy and simplicity or interpretability of the models. In particular, DL models sacrifice the model interpretability for greater performance and end-to-end training [79]. Recent approaches used to gain insights about DL training include visualising the convolutional filters and activation maps or using more advanced, gradient-based methods such as CAM [78] or Grad-CAM [79].

Chapter 4 and chapter 5 employed a gradient-based method, namely Grad-CAM proposed by Selvaraju et al. [79], in order to localise and visualise the input sequences that contributed more to a specific prediction. Grad-CAM uses the gradient information from the last convolutional layers of the CNN to highlight the importance of each neuron for the decision of interest. Therefore, Grad-CAM methods could be used for any type of CNN network, including the ones proposed in the studies presented in chapters 4 and 5. The input of both studies consisted of extracted heartbeats from a long-term ECG recording, thus the regions identified by Grad-CAM as being class-discriminative could be used by medical professionals to understand better the changes in the heartbeat morphology related to either low glucose levels or to CHF. Moreover, visualising the subsequences in the input heartbeat for the low glucose detection, revealed that the proposed model (i.e. CNN) identified different important regions for each individual, therefore advancing the idea that the ECG alterations induced by low glucose levels are person-specific. In contrast to object detection problems, where the point of interest in an image is straightforward, for biomedical problems such as the ones investigated, the underlying physiological process are not yet completely understood, therefore, methods such as Grad-CAM have the potential to reveal unknown, unexplored physiological mechanisms.
7.4 Limitations and Future Work

The studies presented in this thesis produced relevant contributions to the body of knowledge related to ECG analysis using DL methods for healthcare monitoring applications. However, due to time and resource constraints, this research presents some limitations, which represent valuable calls for future research directions.

Firstly, the study presented in chapter 3 identified the recurrent applications of ECG analysis using DL-based methods, the recurrent DL models used for specific tasks and the recurrent ECG preprocessing steps. The results of the analysis performed in chapter 3 were based on data extracted from a limited number of studies, that needed to comply with the defined inclusion/exclusion criteria. Therefore, some studies that used DL methods for ECG analysis might have been excluded due to the violation of either search or inclusion criteria.

Moreover, studies that transform the ECG into different representations such as images, or the RR series instead of the raw ECG might provide promising results and future research directions, however the review in chapter 3 only considered studies that used raw ECG signals as input. Furthermore, due to different protocol and validation methods employed in the reviewed studies, the results comparison of different methods applied on the same dataset was not facilitated. Therefore, future work might include a quantitative analysis of the performance of different DL models and architectures that were employed for the same task on the same dataset. However, to promote this comparison, standard training and testing datasets should be defined, as well as standard validation methods (i.e. intra- or inter- subject). The review presented in chapter 3 revealed a high heterogeneity across the structure of the DL models, therefore, the need for standardising certain hyperparameters was identified.

The studies presented in chapter 4 confirmed the feasibility of building a non-invasive, non-intrusive low glucose detection system based on ECG analysis, recorded in free-living conditions using wearable sensors. A first limitation of the two case studies is given by the relative small number of enrolled participants. Moreover, both case studies presented in section 4.3 and section 4.4 only considered healthy individuals, therefore it was expected that their low glucose events are limited or in some even absent. Future studies should also include type 1 and type 2 diabetic patients considering a wide age range to provide further evidence for supporting the obtained results.

Moreover, the findings in study 3 (section 4.4) regarding the performance improvements of the low glucose detection system when using an unsupervised pre-
training step should also be tested on the models employed in study 2 (section 4.3). Furthermore, the focus of the both of the studies was to investigate the feasibility of a low glucose detection system using only ECG recordings, and not necessarily on obtaining the best performance. Future work can include a broader search over the hyperparameters for improving the current, reported performance.

Other biosignals are known to be affected by hypoglycemia, therefore the integration of multiple sources of information such as heart rate, skin conductance and temperature might improve the accuracy of the proposed system.

Deploying the models on a cloud platform or in a wearable device and developing a phone application for the alarms, to test the system in real-time, represents the next step towards a complete prototype of the proposed system.

The study presented in chapter 5 investigated a CNN-based system for the CHF diagnosis using a benchmark dataset. Although promising results were obtained, more recent, clinical data are required in order to use the proposed solution as a clinical decision support system. Moreover, it should be noted that the CHF patients were diagnosed with severe CHF, therefore it would be of interest to investigate whether the proposed solution can be used for less severe CHF diagnosis and even for early CHF detection. With the advances in wearable sensors, such models can be integrated into everyday wearable devices (e.g. watch, chair back rest, etc.) for disease prevention or monitoring.

The study presented in chapter 6 investigated whether certain HRV features could be inferred from the raw ECG signal. In particular, four features have been tested: AVNN, IHR, SDNN and HF and promising results have been obtained for AVNN and IHR. The study also aimed at determining the minimum number of heartbeats necessary for the accurate estimation of these HRV features. Using a shorter ECG signal for the assessment of certain conditions, could facilitate the implementation of real-time monitoring. Future work might include testing additional HRV features similar to AVNN and IHR. Additionally, using the raw ECG signal in conjunction with DL methods in previous applications that employed HRV features as discriminative variables might represent future research avenues.

Finally, the findings of this thesis confirmed that DL methods applied for ECG analysis can improve and extend the current diagnostic models, might bring new research opportunities and contribute to the development of real-time health monitoring applications.
References


181


Appendix A

Supplementary materials for Chapter 3
<table>
<thead>
<tr>
<th>ID</th>
<th>Authors</th>
<th>Title</th>
<th>Application</th>
<th>Dataset</th>
<th>Method</th>
<th>Preprocessing</th>
<th>Architecture</th>
<th>Results</th>
<th>Year</th>
<th>Source title</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Tanveer M.S., Hasan M.A.</td>
<td>A novel Bayesian deep learning methodology for enhanced foetal cardiac signal mining</td>
<td>Fetal ECG mining</td>
<td>ANA-LSTM</td>
<td>CNN</td>
<td>513 samples, heavy processing, filtering, waveform decomposition</td>
<td>LSTM bidirectional hidden states 128</td>
<td>MAE: 1.1 mmHg SBP: 12 leads: &gt;99%</td>
<td>2019</td>
<td>Biomedical Signal Processing and Control</td>
</tr>
<tr>
<td>6</td>
<td>Kadi S., Tanveer T., Donhi, Ali-E. Ul-Ame M.D., Sekine M., Kanaya S., Huang M.</td>
<td>A novel CNN based framework for ECG quality assessment and Sleep position from a Capacitive ECG measurement</td>
<td>ECG quality assessment &amp; Sleep position detection</td>
<td>PTB &amp; ECG</td>
<td>CNN</td>
<td>128 electrodes were recorded for a 30-minute experiment</td>
<td>CNN: 9 conv layers</td>
<td>Signal quality detection: precision: 97%, recall: 97%</td>
<td>2019</td>
<td>Sensors (Switzerland)</td>
</tr>
<tr>
<td>7</td>
<td>Jangang C., Rawena Kuie Doly O., Peter J.O.</td>
<td>A novel Bayesian deep learning methodology for enhanced foetal cardiac signal mining</td>
<td>Fetal ECG mining</td>
<td>ECOCODINE &amp; DARTY</td>
<td>CNN</td>
<td>nothing specified</td>
<td>CNN: 3 conv layers</td>
<td>MAE: 1.1 mmHg DBP: &gt;99%</td>
<td>2019</td>
<td>Journal of Experimental and Theoretical Artificial Intelligence</td>
</tr>
<tr>
<td>8</td>
<td>Hammad M., Wang K.</td>
<td>Parallel score fusion of ECG and fingerprint for human authentication based on convolutional neural network</td>
<td>Biometrics</td>
<td>PTB and InOut</td>
<td>CNN</td>
<td>nothing specified</td>
<td>CNN: 3 conv layers</td>
<td>Acc: 99% on the fusion of ECG and fingerprint</td>
<td>2019</td>
<td>Computers and Security</td>
</tr>
<tr>
<td>9</td>
<td>Chandra B.S., Satiny C.S., Janas S.</td>
<td>Robust Heartbeat Detection from QRS annotation</td>
<td>Heartbeat Detection</td>
<td>MITDB</td>
<td>CNN</td>
<td>fiducial 300 samples, heartbeat peak signal down sampled to 250 Hz</td>
<td>CNN: 1 conv layer</td>
<td>Acc on PYSQ02014: 94%, Acc on MITDB: 99.92%</td>
<td>2019</td>
<td>IEEE Transactions on Biomedical Engineering</td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Preprocessing</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
</tr>
<tr>
<td>----</td>
<td>---------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>--------------------------------------------------</td>
<td>----------------</td>
<td>------------------------------------------------------------------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
<td>------</td>
<td>------------------</td>
</tr>
<tr>
<td>14</td>
<td>Strachan H., Strawford R.</td>
<td>Detecting and interpreting arrhythmia using convolutional neural networks</td>
<td>Myocardial infarction detection</td>
<td></td>
<td>DWT (Haar)</td>
<td></td>
<td>Sens: 89.3%, Spec: 83.7%</td>
<td>2019 Physiological Measurement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Xia Y., Xie Y.</td>
<td>A novel wearable electrocardiogram classification using convolutional neural networks and active learning</td>
<td>Electrocardiogram classification</td>
<td></td>
<td>CNN-active learning</td>
<td></td>
<td>Acc: 99% on all datasets</td>
<td>2019 IEEE Access</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Preprocessing</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source</td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>-------------</td>
<td>---------</td>
<td>--------</td>
<td>-------------</td>
<td>-------------</td>
<td>---------</td>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>19</td>
<td>Patanè A., Riccabonaschi M.</td>
<td>Calibrating the classifier: Siamese neural network architecture for end-to-end annual recognition from ECG</td>
<td>Emotion recognition</td>
<td>Schizophrenia</td>
<td>ONH/HHHS dataset</td>
<td>non-fiducial filter at aroundf 16 Hz, ECG segments sampled 20 times segments of 1024 intra-subject</td>
<td>CNN: 6 conv layers Input(300,240)[:-1]: (C28x6)-P(max)-C(32x6)[:-1]: (C128x7)-P(max)-C(128x7)-C(128x7)-RNN: 3 RNN layers RNN hidden units: 128, 128, 256 Siamese network</td>
<td>AUC: 86%</td>
<td>2019</td>
<td>Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and lecture Notes in Bioinformatics)</td>
</tr>
<tr>
<td>20</td>
<td>Parunimi M., Castillo R., Pacheco L.</td>
<td>Estimation of the heart rate variability features via recurrent neural networks</td>
<td>HRV inference</td>
<td>SHHS LSTM</td>
<td>inter-subject</td>
<td>LSTM: 1 hidden layer 64 units</td>
<td>NME: MAE S: 5.4%</td>
<td>2019</td>
<td>PMIEC Proceedings</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Van Zaai J., Chételat O., Comoy M., Cailes E.M., Delgado-Gonzalo R.</td>
<td>Classification of cardiac arrhythmias from single lead ECG with a convolutional recurrent neural network</td>
<td>ECG classification</td>
<td>PHYSIS2017 + recorded data</td>
<td>ONH/HHHS dataset</td>
<td>batchnormalized band-pass filter (0.5-40 Hz) downsampled to 2016 Hz normalization on the entire ECG recording input: extract 512 samples with 50% overlap + dataaugmentation Intra-subject</td>
<td>CNN: 7 conv layers Input(512):2x(256x5)-P(max)-RNN: 3 hidden input: input 512, 16 samples 1.28 s</td>
<td>F1 for a subset of the training dataset: 84.9%</td>
<td>2019</td>
<td>BIOSIGNALS 2019 - 12th International Conference on Bio-Inspired Systems and Signal Processing, Proceedings Part of 12th International Joint Conference on Biomedical Engineering</td>
</tr>
<tr>
<td>22</td>
<td>Attia Z.I., Kapa S., Lopez-Jimenez F., Nobilis M., Ledevey D., Salom G., Petitbois P.A., Enriquez-Sarano M., Nasreddy P.A., Mungell F.M., Anitha N.J., Scott C.G., Carter R.E., Friedman P.A.</td>
<td>Screening for cardiac contractile dysfunction using an artificial intelligence-enabled electrocardiogram</td>
<td>LV dysfunction</td>
<td>Mayo Clinic data, 449 380 patients</td>
<td>ONH/HHHS dataset</td>
<td>nothing specified about preprocessing 1024 sampleslead as input &quot;*&quot; in inter-subject</td>
<td>CNN: 7 conv layers, for each 12 lead Input(12x480)-2x(64x5)-P(max)-2x(32x5)-P(max)-C(12x1)-F(7)-F(7)</td>
<td>Acc: 85.7%</td>
<td>2019</td>
<td>Nature Medicine</td>
</tr>
<tr>
<td>23</td>
<td>Ghalwa C., Valyi A.V., Shehabi A.J., Freeman O.C., Peterson F.L., Gundota V.P., Albert O., Atto A., Carter R.E., Anitha N.J., Karpman M., Nairwarsky P.A., Dallan J.L., Friedman P.A.</td>
<td>Development and validation of a deep learning model to screen for hyperkalemia from the Electrocardiogram</td>
<td>hyperkalemia detection</td>
<td>Mayo Clinic data, 449 380 patients</td>
<td>ONH/HHHS dataset</td>
<td>ECG downsampling to 300 Hz 16x ECG 2 and 49d ECG Inter-subject</td>
<td>CNN: 5 conv layers Input(512x1024)-2x(256 or 32 or 16 x16)-FC(2)</td>
<td>AUC: 85.5% to 88.3%</td>
<td>2019</td>
<td>JMAMedical</td>
</tr>
<tr>
<td>24</td>
<td>Hannun A.Y., Rupkar R., Haghpanahi M., Than G.H., Bashir N., Turakhia M.P., Ng A.Y.</td>
<td>Cardiac level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network</td>
<td>ECG classification</td>
<td>53449 patients, 200Hz</td>
<td>ONH/HHHS dataset</td>
<td>1 lead ECG 30min/patient output prediction every 1.28 s 12 days normalization Inter-subject</td>
<td>CNN: 34 conv layers Input(12x480)-2x(64x5)-P(max)-2x(32x5)-P(max)-2x(16x5)-P(max)-C(12x1)-F(7)-F(7)</td>
<td>Acc: 97.8%</td>
<td>2019</td>
<td>Nature Medicine</td>
</tr>
<tr>
<td>25</td>
<td>Attia Z.I., Kapa S., Yao X., Lopez-</td>
<td>Prospective validation of a deep learning electrocardiogram algorithm for the detection of left ventricular systolic dysfunction</td>
<td>LV dysfunction</td>
<td>Mayo Clinic data, 160000 new subjects</td>
<td>ONH/HHHS dataset</td>
<td>nothing specified about preprocessing 1024 sampleslead as input &quot;*&quot; in inter-subject</td>
<td>CNN: 7 layers, for each 12 lead Input(12x480)-2x(64x5)-P(max)-2x(32x5)-P(max)-2x(16x5)-P(max)-C(12x1)-F(7)-F(7)</td>
<td>Acc: 86.5%</td>
<td>2019</td>
<td>Journal of Cardiovascular Electrophysiology</td>
</tr>
<tr>
<td>26</td>
<td>Sikka P., Lustg., Warren F., Mannari D., Marra G.</td>
<td>QRS detection in ECG signal with convolutional neural network</td>
<td>QRS annotation</td>
<td>MITDB ONH</td>
<td>300 samples shifted by 20, 50, 80, 120 samples no filtering Inter-subject</td>
<td>CNN: 4 layers Input(300,240)[:-1]: (C28x6)-P(max)-C(32x6)[:-1]: (C128x7)-P(max)-C(128x7)-P(max)-C(128x7)-P(max)-C(128x7)-P(max)-P(max)-FC(4096)-FC(4096)</td>
<td>Sens 99.95% PPV 97.2%</td>
<td>2019</td>
<td>Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and lecture Notes in Bioinformatics)</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Ghoshhatrami Z., Acar A.</td>
<td>ECG classification using three-level fusion of different feature descriptors</td>
<td>ECG classification</td>
<td>MITDB ONH</td>
<td>fiducial intra-subject noise removed, waveform 6 Classes: 280 sampleshearbeat normalization</td>
<td>CNN: 5 layers Input(128[-1]: (C28x6)-P(max)-C(32x6)-P(max)-C(128x7)-P(max)-C(128x7)-P(max)-P(max)-FC(4096)-FC(4096)</td>
<td>Acc: 98%</td>
<td>2018</td>
<td>Expert Systems with Applications</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Preprocessing</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>-------------</td>
<td>---------</td>
<td>--------</td>
<td>---------------</td>
<td>--------------</td>
<td>---------</td>
<td>------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| 31 | Nguyen M.T., Nguyen B.V., Kim K. | Deep Feature Learning for Sudden Cardiac Arrhythmia Detection in Automated External Defibrillators | Cardiac arrest detection | CUDB & MITMVADB | CNN | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | Scientific Reports 
| 32 | Yihirim O., Tan R.S., Achariyasiri R. | An efficient compression of ECG signals using deep convolutional autoencoders | ECG compression | MITDB | CAE | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | Cognitive Systems Research 
| 33 | Lee J.S., Seo M., Kim S.W., Choi M. | Fusion of 3 CNNs for rapid arrhythmia detection based on convolutional neural networks in noninvasive fetal electrocardiogram | Fetal QRS detection | PHYSIO2013 | CNN | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | International Conference on Frontiers of Signal Processing, ICFSP 2018 
| 34 | Zhang W., Yu L., Ye L., Zheng W. | ECG Signal Classification with Deep Learning for Heart Disease Identification | ECG classification | MITDB | CNN | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | International Conference on Big Data and Artificial Intelligence, BDIA 2018 
| 35 | Xu L., Li T., Chen Y., Chen W. | Personal Identification by Convolutional Neural Network with ECG Signal | Biometrics | 10healthy by during bathing, 10Hz ECG not clear when it was used | CNN | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | 9th international Conference on Information and Communication Technology Convergence: ICT Convergence powered by Smart Intelligence, ICTC 2018 
<p>| 36 | Li Y., Pang Y., Wang J., U X | Patient-specific ECG classification by deep CNN not generated to dedicated | ECG classification | MITDB | CNN | intra-subject | 3 max pooling layers fusion of 3 CNNs (input size 256×256) (P=mean)-C(1024)-C(1024) (P=mean)-C(1024)-C(1024) (P=mean) Spec: 97.67% | Acc: 99.2% | 2018 | Neurocomputing |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Authors</th>
<th>Title</th>
<th>Application</th>
<th>Dataset</th>
<th>Method</th>
<th>Preprocessing</th>
<th>Architecture</th>
<th>Results</th>
<th>Year</th>
<th>Source Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>37</td>
<td>Guan J., Lui R., Li W., Wang J., Xie G.</td>
<td>Automated dynamic electrocardiogram noise reduction using multilayer LSTM network</td>
<td>ECG noise reduction</td>
<td>MITDB &amp; NTDDB</td>
<td>LSTM</td>
<td>3 hidden layers</td>
<td>RMSE between 0.025-0.037</td>
<td>2018</td>
<td>IEEE International Conference on Biomedical Engineering and Informatics</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>Fan X., Yao Q., Cai Y., Mi et al., Sun F., Y.</td>
<td>Multilevel Fusion of Deep Convolutional Neural Networks for Screening Arrhythmia Detection from Single Lead ECG Recordings</td>
<td>arrhythmia detection</td>
<td>PHYSIO2017</td>
<td>CNN</td>
<td>13 conv layers</td>
<td>Acc: 91.3% - 95.20% for 13 to 17 classes</td>
<td>2018</td>
<td>Computers in Biology and Medicine</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>Yi H., Li H., Tan K., Acharya U.R.</td>
<td>Arrhythmia detection using deep convolutional neural network with long duration ECG signals</td>
<td>ECG deasification</td>
<td>MITDB</td>
<td>CNN</td>
<td>non-fiducial</td>
<td>Acc: 98.1% Sens: 97.5%</td>
<td>2018</td>
<td>Computers in Biology and Medicine</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>Oh S.L., Ng J.K., Tan R.S., Acharya U.R.</td>
<td>Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heartbeat</td>
<td>ECG deasification</td>
<td>MITDB &amp; CHINA</td>
<td>CNN+LSTM</td>
<td>1 hidden layer</td>
<td>Acc: 98.1% Sens: 97.5%</td>
<td>2018</td>
<td>Computers in Biology and Medicine</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>Zhou X., Zhu X., Nakamura K., Makris M.</td>
<td>Premature Ventricular Contraction Detection from Ambulatory ECG Using Recurrent Neural Networks</td>
<td>PVC detection</td>
<td>MITDB</td>
<td>RNN</td>
<td>6 hidden layers, 300 units not clear what is the input</td>
<td>Acc: 96% and 99%</td>
<td>2018</td>
<td>Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBS</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Chen Y., Chen W.</td>
<td>Finger ECG based Two phase Authentication Using ID Convolutional Neural Networks</td>
<td>Biometrics</td>
<td>50 subjects</td>
<td>CNN</td>
<td>5 conv layers</td>
<td>FRR and FPR ~ 10%</td>
<td>2018</td>
<td>Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society EMBS</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Preprocessing</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
</tr>
<tr>
<td>----</td>
<td>--------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------</td>
<td>---------------------</td>
<td>-------------------------</td>
<td>--------------------------------</td>
<td>-----------------------------</td>
<td>--------------------------</td>
<td>------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>46</td>
<td>Haradal S., Hayashi H., Uchida S.</td>
<td>Biosignal Data Augmentation based on Generative Adversarial Networks</td>
<td>Data augmentation &amp; EEG classification</td>
<td>ECG 200(UCI) &amp; Epileptic seizure recognition data set (UCI)</td>
<td>GAN</td>
<td>ECG 2019: 96 samples</td>
<td>G-LSTM: 3 layers and 200 units</td>
<td>Acc: 75%</td>
<td>2018</td>
<td>Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS</td>
</tr>
<tr>
<td>47</td>
<td>Rajan D., Thagapalan J.J.</td>
<td>A Generative Model Approach to Limited Channel ECG Classification</td>
<td>ECG classification</td>
<td>PTB</td>
<td>AE</td>
<td>3 channel input - generate 12 channel of E5% ECG downsampled to 510 samples normalization</td>
<td>E-LSTM: 2 layer, with 64 Hidden units</td>
<td>Acc: 80%</td>
<td>2018</td>
<td>Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS</td>
</tr>
<tr>
<td>48</td>
<td>Ansari S., Gryak J., Najarian K.</td>
<td>Noise Detection in Electrocardiography Signal for Robust Heart Rate Variation Analysis: A Deep Learning Approach</td>
<td>ECG quality assessment</td>
<td>MITDB &amp; NSTDB</td>
<td>CNN</td>
<td>3s ECG butter filter 3 Hz downsampled to 128Hz normalization subject</td>
<td>CNN:4 conv layers Input([384x32x7])-P(max)-C[32x7]-P(max)-C[64x7]-P(max)-C[64x7]-P(max)</td>
<td>AUC: 95%</td>
<td>2018</td>
<td>Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS</td>
</tr>
<tr>
<td>49</td>
<td>Warrick P.A., Nabhan Hamu M.</td>
<td>Ensembling convolutional auto-encoder short-term memory networks for electrocardiogram rhythmia detection</td>
<td>ECG classification</td>
<td>PHYSIO2017</td>
<td>CNN</td>
<td>9000 samples as input</td>
<td>CNN: 1 conv layer ensemble of 10 LSTM 3 Hidden layers, 100 Hidden units Input(5000-37)-P(max) LSTM(100)-LSTM(100)-LSTM(100)</td>
<td>F1: 82%</td>
<td>2018</td>
<td>Physical Signal Measurement</td>
</tr>
<tr>
<td>50</td>
<td>Sodemann P., Volmer M., Nath N., Kaderali L.</td>
<td>A convolutional deep network for ECG annotation as the basis for classification of cardiac rhythms</td>
<td>QRS annotation &amp; ECG classification</td>
<td>PWAVE &amp; PHYSIO2017</td>
<td>CNNfeatures</td>
<td>3s ECG butter filter 3 Hz downsampled to 300Hz DWT for filtering band pass filter 40-60 samples normalization [1-1] subject</td>
<td>CNN: Scan layers Input([5000]-C[64x3]-P(max)-C[32x3]-C[16x3]-P(max)-C[16x3]-P(max)-C[16x3]-P(max)-Flatten)</td>
<td>detection of R spikes: 99% P.T. detection within 50ms tolerance: 92% and 88%</td>
<td>2018</td>
<td>Physical Signal Measurement</td>
</tr>
<tr>
<td>52</td>
<td>Sayantan G., Kim K.P., Kadambali V.K.</td>
<td>Classification of ECG beats using deep belief network and active learning</td>
<td>ECG classification</td>
<td>MITDB &amp; MTSDB</td>
<td>DBN+features</td>
<td>median filter 12th order low pass filter 30Hz cut off 50 samples/input + 4 feature values</td>
<td>DBN: 2 Hidden layers</td>
<td>after expert queries: 99% accuracy on MITDB 97.5% for SVDB and 98.6% for VDB on SVDB</td>
<td>2018</td>
<td>Medical and Biological Engineering and Computing</td>
</tr>
<tr>
<td>53</td>
<td>Xiong Z., Noah M.P., Cheng E., Feiweis N.V., Silve M.K., Zhao J.</td>
<td>ECG signal classification for the detection of cardiac arrhythmias using a convolutional recurrent neural network</td>
<td>atrial fibrillation detection</td>
<td>PHYSIO2017</td>
<td>CNNHPNN</td>
<td>5s ECG</td>
<td>CNN: 8 core layers Input([15000]-200x300)-C[14x5]-C[14x5]-C[14x5]-FC(1024)-3x(BN)(12)-RNN: 3Hidden layers</td>
<td>F1: 86.4%</td>
<td>2018</td>
<td>Physiological Measurement</td>
</tr>
<tr>
<td>54</td>
<td>Xu H., Mak M., Cheung C.</td>
<td>Toward End-to-End ECG Classification with Raw Signal Extraction and Deep Neural Networks</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN</td>
<td>fiducial heartbeat segmentation and alignment 41 samples/heartbeat</td>
<td>DBN: 3 layers for pretraining MLP: 3 layers 100 neurons in each SONE for visualization</td>
<td>Overall Acc 94.7%</td>
<td>2018</td>
<td>IEEE Journal of Biomedical and Health Informatics</td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>---------</td>
<td>-----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------</td>
<td>------</td>
<td>------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>Golrizkhatami Z., Taheri S., Acan A.</td>
<td>Multi-scale features for heartbeat classification using directed acyclic graph CNN</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN</td>
<td>4 convolution layers: Input(C(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32))</td>
<td>Acc: 97.15%</td>
<td>2018</td>
<td>Applied Artificial Intelligence</td>
<td></td>
</tr>
</tbody>
</table>
| 56 | Plesinger F., Najafizada M., Valcarce J., Holamek J., Jurak P. | Parallel use of a convolutional neural network and a bagged-tree ensemble for the classification of Holter ECG | ECG classification                                                        | PHYSIO2017 | CNN                                                                | 2 convolution layers: Input(x(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | F1: 94%  
|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | Accuracy: 92%  
|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | Quality: 74%                                                       |      | Physiological Measurement                                                   |
| 57 | Li J., Yi Y., Liang L., Liu L., Xu T.         | A spatial pyramid pooling-based deep convolutional neural network for the classification of electrocardiograms beats | ECG classification                                                        | MITDB  | ONH/SPY                                                               | 2 convolution layers: Input(x(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Acc: 94%                                                              | 2018 | Applied Sciences (Switzerland)                                               |
| 58 | Liu W., Zhang M., Zhang Y., Liu Y., Huang Q., Chang S., Wang H., He J. | Real-Time Multi Lead Convolutional Neural Network for Myocardial Infarction detection | PTB                                                                        | ONH/SPY| CNN                                                                  | 3 convolution layers: Input(x(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Acc: 100%  
|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | Sensitivity: 95%  
|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | Specificity: 97.57%                                                     |      | IEEE Journal of Biomedical and Health Informatics                              |
| 59 | Marugan O., Bhattacharyya V., Ramkutty P., Joseph J., Shankaranarayanan S.M., Sirovnikou M. | ECGNet: Deep Network for Arrhythmia Classification | ECG classification                                                        | PTB     | CNN                                                                  | 3 convolution layers: Input[x(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Acc: 97.6%                                                              | 2018 | IEEE International Symposium on Medical Measurements and Applications, Proceedings |
| 60 | Liu W., Huang Q., Chang S., Wang H., He J. | Multiple-feature-branch convolutional neural network for myocardial infarction diagnosis using electrocardiographic signal | Myocardial infarction detection                                            | PTB     | CNN                                                                  | 3 convolution layers: Input(x(8x32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Class based: Acc: 98.82%  
|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | Patient-specific Acc: 94.82%                                             |      | Biomedical Signal Processing and Control                                     |
| 61 | Kashuea M., Faezi S., Saradabhak M.           | ECG heartbeat classification: A deep transferable representation of a heartbeat classification | ECG classification                                                        | PTB     | CNN                                                                  | 11 convolution layers: Input[6x128x16]-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Acc: 95.9%                                                              | 2018 | Proceedings - 2018 IEEE International Conference on Healthcare Informatics, ICHI 2018 |
| 62 | Yang Y., Yu L., Ji Q., Wu L., He B.          | Localization of Origins of Premature Ventricular Contraction by Means of Convolutional Neural Network from 12-Lead ECG | PVC localization                                                        | 9 patients 2kHz 10-30 minutes | CNN                                                                  | 2 convolution layers: Input[x(32x16)-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Segment PVC Acc: 78%  
<p>|     |                                              |                                                                      |                                                                            |         |                                                                      |                                                                            | ECG-End Acc: 93%                                                        |      | IEEE Transactions on Biomedical Engineering                                  |
| 63 | Wu M.-H., Chang E.J., Chu T.H.               | Personalizing a Deep Convolutional ECG Heartbeat Classification for Arrhythmia Detection: A Deep Learning Approach | ECG classification                                                        | MITDB  | CNN                                                                  | 4 convolution layers: Input[6x128x16]-P(max)-C(32x2x3)-P(max)-C(16x4)-P(max)-C(8x2x2)-P(max)-C(4x2x2)-P(max)-C(2x2x2)-P(max)-C(1x2x2)-P(max)-C(1x1x1)-FC(32)) | Acc for N and MVB ~ 100%                                               | 2018 | Proceedings - IEEE LIT Conference on Multimodal Information Processing and Retrieval, MMMN 2018 |</p>
<table>
<thead>
<tr>
<th>ID</th>
<th>Authors</th>
<th>Title</th>
<th>Application</th>
<th>Dataset</th>
<th>Method</th>
<th>Preprocessing</th>
<th>Architecture</th>
<th>Results</th>
<th>Year</th>
<th>Source title</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>Zhang Y., Lee X.</td>
<td>Deeply learned electrocardiogram representations are robust</td>
<td>Biometrics</td>
<td>PTB &amp; MITDB &amp; Fantasia &amp; CIVH &amp; THUADT (private)</td>
<td>CNN</td>
<td>intra-subject 4th or 6th butterworth downsampling to 25 Hz 136 ECG windows</td>
<td>CNN: 2 parallel conv layers, each with 3 convs of different filter sizes Input: [160] 8x1000 samples of ECG CNN: 2 parallel conv layers, each with 3 convs of different filter sizes Input: [160] 8x1000 samples of ECG</td>
<td>Acc 98.49%</td>
<td>2018</td>
<td>IEEE-EMBS 2017 - 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery</td>
</tr>
<tr>
<td>65</td>
<td>Urtanasan E., Park J.-I., Lee K.-J.</td>
<td>Multiclass classification of obstructive sleep apnea/hypopnea based on a convolutional neural network from a single lead electrocardiogram</td>
<td>OSA detection</td>
<td>86 subjects</td>
<td>CNN</td>
<td>intra-subject bandpass filter (5-11 Hz) 10s ECG</td>
<td>CNN: 6 conv layers Input: [160] 8x200 samples of ECG CNN: 6 conv layers Input: [160] 8x200 samples of ECG</td>
<td>FL: 93% train error 10 8.7% test Error 10: 98.9%</td>
<td>2018</td>
<td>International Conference on Biomedical and Health Informatics, BHI 2018</td>
</tr>
<tr>
<td>68</td>
<td>Taj B., Khan A.D.C., Shimmurungam S.</td>
<td>False-Alarm Reduction in Fetal Arrhythmia Detection Using Deep Belief Networks</td>
<td>Biometrics</td>
<td>MITAFDB &amp; NSTDB for training and noise</td>
<td>CNN</td>
<td>intra-subject bandpass filter (5-15 Hz) 200 Hz inter-subject</td>
<td>CNN: 10E layers, 10,000 hidden neurons 3x3 RBM unsupervised classifier</td>
<td>Acc without noise: 87% Acc with 2008 noise 81%</td>
<td>2018</td>
<td>IEEE Transactions on Instrumentation and Measurement</td>
</tr>
<tr>
<td>70</td>
<td>He Z., Zhang X., Cao Y., Lu Z., Zhang B., Wang X.</td>
<td>Heart rate estimation based on deep learning shortest algorithm for detecting atrial arrhythmias using multi-channel devices</td>
<td>Biometrics</td>
<td>MITDB</td>
<td>CNN</td>
<td>intra-subject 1 or 2 ECG downsampling to 200 Hz 2x100 samples 200 Hz inter-subject</td>
<td>CNN: 3x3 layers Input: [160] 8x200 samples of ECG CNN: 3x3 layers Input: [160] 8x200 samples of ECG</td>
<td>FL: 8.78% for 1s ECG Acc: 98.8% for 2s ECG</td>
<td>2018</td>
<td>Sensors (Switzerland)</td>
</tr>
<tr>
<td>71</td>
<td>Jin S., Deng J.</td>
<td>Intelligent Health Vessel ABC-DE: An Electrocardiogram Cloud Computing Service</td>
<td>Biometrics</td>
<td>CIRCCD &amp; CloudIOT</td>
<td>CNN + ensembles</td>
<td>intra-subject bandpass filter (0 60Hz) downsampling to 200 Hz 4x100 samples 200 Hz inter-subject</td>
<td>CNN: 3 conv layers Input: [160] 8x200 samples of ECG CNN: 3 conv layers Input: [160] 8x200 samples of ECG</td>
<td>FL: 97.9% Sens: 97.9% Spec: 97.9% CIRCCD Sens: 96.4% Spec: 98.9%</td>
<td>2018</td>
<td>IEEE Transactions on Cloud Computing</td>
</tr>
<tr>
<td>72</td>
<td>Abirashmi H., Campbell N., Han C., Casale R., Zhou X.</td>
<td>QRS-T localization in ECG using deep learning</td>
<td>Biometrics</td>
<td>QTDB</td>
<td>CNN</td>
<td>intra-subject 300 samples, heartbeats with padding median filter -drift</td>
<td>CNN: 2 conv layers Input: [160] 8x200 samples of ECG CNN: 2 conv layers Input: [160] 8x200 samples of ECG</td>
<td>Acc sample error 4: 94.2% sample error 5: 98.4% sample error 10: 98.9%</td>
<td>2018</td>
<td>2018 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2018</td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----------------------</td>
<td>----------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
<td>------------------</td>
<td>----------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------</td>
<td>----------------------------</td>
<td></td>
</tr>
<tr>
<td>74</td>
<td>Xiang Y., Luo J., Zhu T., Wang S., Xiang X., Meng J.</td>
<td>ECG-Based heartbeat classification using a two-level convolutional neural network and DNN internal difference</td>
<td>ECG classification</td>
<td>MITDB &amp; INCARTDB</td>
<td>CNN</td>
<td>心电图特征提取和分类</td>
<td>CNN: 2 conv layers input[64x32] x [64x32] x 2 len = 100 Spec: 99%  Pre-processing: MITDB</td>
<td>MBT Acc: 97.8%  INCART Acc: 97.8%</td>
<td>2018</td>
<td>EUR Transactions on Information and Systems</td>
</tr>
<tr>
<td>80</td>
<td>Lui H.W., Chow K.L.</td>
<td>Multiclass classification of myocardial infarction with convolutional and recurrent neural networks for portable ECG devices</td>
<td>Multiclass classification</td>
<td>PTB &amp; PMQ</td>
<td>CNN</td>
<td>心电图特征提取和分类</td>
<td>CNN: 8 conv layers input[512x8x8] x [10x10x10] x 32 max pooling [32x32x32] x [16x16x16] x 32 Spec: 99.4%  Pre-processing: MITDB</td>
<td>MBT Acc: 97.4%  INCART Acc: 97.7%  Fl: 94.0%</td>
<td>2018</td>
<td>Informatics in Medicine</td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>-------------------------------</td>
<td>------------------------------------------------------</td>
<td>-----------------------------</td>
<td>---------------</td>
<td>-------------------------</td>
<td>---------------------------------------------------</td>
<td>-----------------------------</td>
<td>------</td>
<td>-------------------------------------</td>
<td></td>
</tr>
<tr>
<td>81</td>
<td>Swapna G., Soman K.P., Vinayakumar R.</td>
<td>Automated detection of cardiac arrhythmia using deep learning techniques</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN &amp; LSTM &amp; ONH</td>
<td>fiducial no filtering, normalization input: 381 samples 2 classes intra-subject</td>
<td>CNN: up to 3 layers input(361)-C(32x2)-C(64x5)-C(128x5)-F(2) OR: Acc: 83.7</td>
<td>2018</td>
<td>Procedia Computer Science</td>
<td></td>
</tr>
<tr>
<td>82</td>
<td>Singh S., Pandey S.K., Power U., Janghel R.R.</td>
<td>Classification of ECG Arrhythmia using Recurrent Neural Networks</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>LSTM</td>
<td>2 classes intra-subject</td>
<td>LSTM: 2 layers input: 64, 256, 100 hidden units Acc: 88.1%</td>
<td>2018</td>
<td>Procedia Computer Science</td>
<td></td>
</tr>
<tr>
<td>83</td>
<td>Urtnasan E., Park J.-U., Lee K.-J.</td>
<td>Automatic detection of sleep-disordered breathing events using recurrent neural networks from an electrocardiogram signal</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>LSTM</td>
<td>6 classes normal, apnea, hypo-see inter-subject 2000 input samples input: ECG + RR interval, heart rate, respiration inter-subject</td>
<td>LSTM: 4 layers input: 120, 100, 80, 60, 40, 20 Acc: 98%</td>
<td>2018</td>
<td>Neural Computing and Applications</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>Lodhi A.M., Qureshi A.N., Sharif U., Ashiq Z.</td>
<td>A novel approach using voting from ECG leads to detect myocardial infarction</td>
<td>Myocardial infarction detection</td>
<td>PTB</td>
<td>CNN</td>
<td>fiducial 651 samples/hb normalisation no other pre-processing 12 leads voting</td>
<td>CNN: 4 conv layers input(651)-C(3x102)-P(max)-C(10x24)-P(max)-C(10x11)-P(max)-C(10x9)-P(max)-C(10x7)-FC(2) Individual test Acc: 88% with at least 8 leads positive 91.8%</td>
<td>Se92.2 % and Pp 96.1 % for VEC MITDB exceed DB: Acc: 91.7% - 97.8%</td>
<td>2018</td>
<td>Advances in Intelligent Systems and Computing</td>
</tr>
<tr>
<td>86</td>
<td>Qiu Y., Huang K., Yao F., Shen H.</td>
<td>A segment-wise reconstruction method based on bidirectional long short term memory for Power line interference suppression</td>
<td>Power line suppression</td>
<td>MITDB</td>
<td>LSTM</td>
<td>resample to 600Hz [-1, 1] normalisation 1 min ECG input: 3 channel wavelet and 2 median filters input site: 3 types RR vector: pre-RR, past RR, mean of Rb RR</td>
<td>LSTM + real data LSTM input: 32 units SNR improvement more than 7dB 0.329 Compared to existing technique (IIR notch filter)</td>
<td>2018</td>
<td>Biocybernetics and Biomedical Engineering</td>
<td></td>
</tr>
<tr>
<td>87</td>
<td>Donald création R., Muñoz E., Pari V., Saiz R., Scott P.</td>
<td>Deep ECG: Convolutional Neural networks for ECG biometric recognition</td>
<td>Biometrics</td>
<td>IDEAL</td>
<td>CNN</td>
<td>IIR filter 3rd order HIP filter - 8 Hb from a 12s ECG resample for 300 Hz inter-subject</td>
<td>CNN: 6 conv layers input(361)-C(32x5)-P(max)-C(128x5)-P(max)-C(256x5)-C(256x8)-FC(2) PTB Acc: 100%</td>
<td>2018</td>
<td>Pattern Recognition Letters</td>
<td></td>
</tr>
<tr>
<td>88</td>
<td>Li D., Zheng J., Zhang Q., Wei X.</td>
<td>Classification of ECG signals based on 1-D convolution neural network</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN</td>
<td>wavelet filtering 201 samples/hb downsampling 130 intra-subject 5 classes</td>
<td>CNN: 2 conv layers input(130)-C(32x8)-P(max)-C(128x7)-P(max)-C(256x5)-FC(2) Acc: 97.5%</td>
<td>2018</td>
<td>2017 IEEE 19th international Conference on e-Health Networking, Applications and Services, Healthcom 2017</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source/Title</td>
<td></td>
</tr>
<tr>
<td>----</td>
<td>----------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>---------------------------------------</td>
<td>-----------------------</td>
<td>-------------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>------</td>
<td>--------------</td>
<td></td>
</tr>
<tr>
<td>89</td>
<td>Kiranyaz S., Ince T., Gabr A.</td>
<td>Personalized Monitoring and Advance Warning System for Cardiac Arrhythmias</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN</td>
<td>intra-subject abnormal beat synthesis is fiducial desmeval 128 sample/60s no normalization</td>
<td>2017</td>
<td>IEEE Transactions on Biomedical Engineering</td>
<td></td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>Acharya U.R., Fujita H., Oh S.L., Haginawa Y., Tan J.H., Adam M.</td>
<td>Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals</td>
<td>Myocardial infarction detection</td>
<td>PTB</td>
<td>CNN</td>
<td>intra-subject 651 sample/60s intra-subject filtering: daubechies wavelet (both with and without) z normalization</td>
<td>2017</td>
<td>Information Sciences</td>
<td></td>
<td></td>
</tr>
<tr>
<td>91</td>
<td>Sarlija M., Jurisic F., Popovic S.</td>
<td>A convolutional neural network trained approach to QRS detection</td>
<td>QRS annotation</td>
<td>MITDB</td>
<td>CNN</td>
<td>inter-subject 200 sample/60s intra-subject filtering: wavelet filter z normalization intra-subject</td>
<td>2017</td>
<td>Computers in Biology and Medicine</td>
<td></td>
<td></td>
</tr>
<tr>
<td>94</td>
<td>Acharya U.R., Fujita H., Uh O.S., Adam M., Tan J.H., Ovcar K.</td>
<td>Automated detection of coronary artery disease using different durations of ECG segments with convolutional neural network</td>
<td>Coronary artery disease diagnosis</td>
<td>Fantasia &amp; INCAWDB</td>
<td>CNN</td>
<td>non-fiducial 2s and 5s ECG intra-subjects DWT, daubechies wavelet upsampled to 275 Hz z normalization</td>
<td>2017</td>
<td>Knowledge-Based Systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>96</td>
<td>Acharya U.R., Fujita H., Uh O.S., Haginawa Y., Tan J.H., Adam M.</td>
<td>Automated detection of arrhythmias using different intervals of ECG segments with convolutional neural network</td>
<td>ECG classification</td>
<td>MITDB &amp; MITDB &amp; CUBIC</td>
<td>CNN</td>
<td>MITDB downsparse to 250 Hz daubechies wavelet filter z normalization non-fiducial 2s and 5s ECG 4 classes</td>
<td>2017</td>
<td>Information Sciences</td>
<td></td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>Gagnia A., Majumdar A., Ward R.</td>
<td>Semi-supervised Stacked Autoencoder for Biomedical Signals Re-Training</td>
<td>ECG reconstruction &amp; ECG classification</td>
<td>MITDBaug &amp; SAE</td>
<td>CNN</td>
<td>MITDB downsparse to 255 Hz normalization 1x ECG(295 samples) both reconstruction and classification in the same optimization step 3 classes</td>
<td>2017</td>
<td>IEEE Transactions on Biomedical Engineering</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>Authors</td>
<td>Title</td>
<td>Application</td>
<td>Dataset</td>
<td>Method</td>
<td>Preprocessing</td>
<td>Architecture</td>
<td>Results</td>
<td>Year</td>
<td>Source title</td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
<td>-------</td>
<td>-------------</td>
<td>---------</td>
<td>--------</td>
<td>---------------</td>
<td>--------------</td>
<td>---------</td>
<td>------</td>
<td>--------------</td>
</tr>
<tr>
<td>98</td>
<td>Salkhor M., Kuo C.-C.J.</td>
<td>ECG-based biometrics using recurrent neural networks</td>
<td>Biometrics</td>
<td>ID &amp; MITDB</td>
<td>RNN</td>
<td>natural</td>
<td>250 samples/subject, 8 sample normalisation on certain number of beats is the RNN input</td>
<td>LSTM: 8 hidden layers, 200 hidden units</td>
<td>Acc: 100% with 9 input beats</td>
<td>2017</td>
</tr>
<tr>
<td>99</td>
<td>Zhou F.-X., Jin L.-P., Dong J.</td>
<td>Premature ventricular contraction detection combining deep neural networks and rules inference</td>
<td>PVC detection</td>
<td>MITDB &amp; CODE</td>
<td>CNN + LSTM</td>
<td>performing the two datasets MITDB and CODE</td>
<td>LSTM: 1 or 2 layers</td>
<td>Acc: 98.03%</td>
<td>2017</td>
<td>Artificial Intelligence in Medicine</td>
</tr>
<tr>
<td>100</td>
<td>Andrade M., Carr D., Pimentel M.A.F., Mahdi A., De Vos M.</td>
<td>Computing feature-based classifiers and convolutional neural networks to detect arrhythmias from short segments of ECG</td>
<td>ECG classification</td>
<td>PHYSIO2017 CNN</td>
<td>-</td>
<td>1 min ECG</td>
<td>4 classes</td>
<td>Acc: 98%</td>
<td>2017</td>
<td>Computing in Cardiology</td>
</tr>
<tr>
<td>101</td>
<td>Ince T., Zabih M., Karayba S., Gobous M.</td>
<td>Learned vs. hand-designed features for ECG beat classification: A comprehensive study</td>
<td>ECG classification</td>
<td>MITDB</td>
<td>CNN</td>
<td>natural</td>
<td>128 samples/subject, either 1 or 3 beats as input into sub-net</td>
<td>CNN: 2 conv layers</td>
<td>~98% Acc for both VEB and SVEB</td>
<td>2017</td>
</tr>
<tr>
<td>103</td>
<td>Ghiasi S., Abdollahpour M., Madsen H., Khaki K., Ghaffari A.</td>
<td>Atrial fibrillation detection using feature-based algorithm and deep convolutional neural network</td>
<td>Atrial fibrillation detection</td>
<td>PHYSIO2017 CNN</td>
<td>CNN</td>
<td>natural</td>
<td>4 classes</td>
<td>Input: list of beats(0) &gt; 30 convolution to 300 Hz, median filter, [L-1] normalisation, 701 samples/subject</td>
<td>CNN: 4 conv layers</td>
<td>FL: 71%</td>
</tr>
<tr>
<td>104</td>
<td>Warnick P., Henss M.N.</td>
<td>Cardiac arrhythmia detection from ECG: combining convolutional and long short-term memory networks</td>
<td>ECG classification</td>
<td>PHYSIO2017 LSTM</td>
<td>CNN</td>
<td>non-natural</td>
<td>30s ECG with padding for 1 000 samples</td>
<td>LSTM: 3 layers</td>
<td>Acc: 83.3%</td>
<td>2017</td>
</tr>
</tbody>
</table>

Figure A.0: Table presenting details regarding the DL architectures and preprocessing steps for the 106 shortlisted papers included in the review.
Appendix B

Supplementary materials for Chapter 4
B.1 Case study 1

(a) Subject 1, CNN, Test 1

(b) Subject 1, CNN+RNN, Test 1

(c) Subject 1, CNN, Test 2

(d) Subject 1, CNN+RNN, Test 2

(e) Subject 1, CNN, Test 3

(f) Subject 1, CNN+RNN, Test 3

207
Figure B.1: Subject 1. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.
(a) Subject 2, CNN, Test 1

(b) Subject 2, CNN+RNN, Test 1

(c) Subject 2, CNN, Test 2

(d) Subject 2, CNN+RNN, Test 2

(e) Subject 2, CNN, Test 3

(f) Subject 2, CNN+RNN, Test 3
(g) Subject 2, CNN, Test 4
(h) Subject 2, CNN+RNN, Test 4
(i) Subject 2, CNN, Train 1
(j) Subject 2, CNN+RNN, Train 1
(k) Subject 2, CNN, Train 2
(l) Subject 2, CNN+RNN, Train 2
Figure B.2: Subject 2. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.
(i) Subject 3, CNN, Train 1

(j) Subject 3, CNN+RNN, Train 1

(k) Subject 3, CNN, Train 2

(l) Subject 3, CNN+RNN, Train 2

(m) Subject 3, CNN, Train 3

(n) Subject 3, CNN+RNN, Train 3
Figure B.3: Subject 3. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.
(c) Subject 4, CNN, Test 2

(d) Subject 4, CNN+RNN, Test 2

(e) Subject 4, CNN, Test 3

(f) Subject 4, CNN+RNN, Test 3

(g) Subject 4, CNN, Test 4

(h) Subject 4, CNN+RNN, Test 4
(i) Subject 4, CNN, Train 1
(j) Subject 4, CNN+RNN, Train 1
(k) Subject 4, CNN, Train 2
(l) Subject 4, CNN+RNN, Train 2
(m) Subject 4, CNN, Train 3
(n) Subject 4, CNN+RNN, Train 3
Figure B.4: Subject 4. Hypoglycemia detection during the recorded nights, using the heartbeat majority voting in a 10-minutes ECG segment.

(o) Subject 4, CNN, Train 4

(p) Subject 4, CNN+RNN, Train 4
B.2 Case study 2

Figure B.5: Subject 1. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.6: Subject 2. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.7: Subject 3. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.8: Subject 4. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.9: Subject 5. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.10: Subject 6. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.
Figure B.11: Subject 7 and Subject 8. Predicted glucose levels for 5-minutes ECG intervals where each point corresponds to a 5-minute ECG interval.