Degree Project in Technology
Second cycle, 30 credits

Straight to the Heart:
Classification of Multi-Channel ECG-signals using MiniROCKET

STEFAN CHRISTIANSSON
Straight to the Heart: Classification of Multi-Channel ECG-signals using MiniROCKET

STEFAN CHRISTIANSSON
Abstract

Machine Learning (ML) has revolutionized various domains, with biomedicine standing out as a major beneficiary. In the realm of biomedicine, Convolutional Neural Networks (CNNs) have notably played a pivotal role since their inception, particularly in applications such as time-series classification. Deep Convolutional Neural Networks (DCNNs) have shown promise in classifying electrocardiogram (ECG) signals. However, their deep architecture leads not only to risk for over-fitting when insufficient data is at hand, but also to large computational costs. This study leverages the efficient architecture of Mini-ROCKET, a variant of CNN, to explore improvements in ECG signal classification at Getinge. The primary objective is to enhance the efficiency of the Electrical Activity of the Diaphragm (Edi) catheter position classification compared to the existing Residual Network (ResNet) approach.

In the Intensive Care Unit (ICU), patients are often connected to mechanical ventilators operating based on Edi catheter-detected signals. However, weak or absent EMG signals can occur, necessitating ECG interpretation, which lacks the precision required for optimal Edi catheter placement. Clinicians have long recognized the challenges of manual Edi catheter positioning. Currently, positioning relies on manual interpretation of electromyography (EMG) and ECG signals from a 9-lead electrode array. Given the risk for electrode displacement due to patient movements, continuous monitoring by skilled clinicians is essential.

This thesis demonstrates the potential of Mini-ROCKET in addressing these challenges. By training the model on Getinge’s proprietary ECG patient dataset, the study aims to measure improvements in computational cost, accuracy, and user value as compared to previous work with Edi-catheter positioning at Getinge. The findings of this research hold significant implications for the future of ECG signal classification and the broader application of Mini-ROCKET in medical signal processing.

Keywords

MiniROCKET, Time-series analysis, Multi-variate, Classification, Convolutional Neural Network
Sammanfattning

Maskinlärning har revolutionerat många områden, varav biomedicin som visat enorm utveckling. Inom biomedicin har konvolutionella neurala nätverk (CNNs) gjort stor positiv påverkan, särskilt inom tillämpningar som tidsserieklassificering. Djupa konvolutionella neurala nätverk (DCNNs) har visat lovande resultat inom elektrokardiogram (EKG) klassificering. Deras djupa arkitektur leder dock inte bara till risk för överanpassning med begränsad data till handa, utan även till betydliga beräkningskostnader. Denna studie utnyttjar den effektiva arkitekturen av Mini-ROCKET, en variant av CNN, för att utforska förbättringar i EKG-signal klassificering på Getinge. Huvudmålet är att förbättra effektiviteten av Edi kateterpositions klassificering jämfört med den befintliga Residual Network (ResNet) metoden.


Nyckelord

MiniROCKET, Tidsserieanalys, Multivariat, Klassifikation, Convolutional Neural Network
Abstract
Acknowledgments

I would like to thank my supervisors Erik Fransén (KTH) and Fredrik Jalde (Getinge MCC AB) for their support and guidance in this research.

Stockholm, November 2023
Stefan Christiansson
# Contents

1 Introduction  
1.1 Background  ............................................. 2  
1.2 Problem and Research Questions ......................... 5  
1.3 Purpose .................................................... 6  
1.4 Goals ....................................................... 6  
1.5 Research Methodology ...................................... 7  
1.6 Ethical and Societal Aspects ............................... 7  
1.7 Delimitations .............................................. 8  
1.8 Structure of the thesis  ................................... 8  

2 Background  
2.1 Time-Series Classification ................................. 10  
2.2 ROCKET ..................................................... 11  
2.3 Mini-ROCKET ............................................... 11  

3 Methods  
3.1 Research Process .......................................... 12  
3.2 ROCKET ..................................................... 13  
3.2.1 Kernels .................................................... 13  
3.2.2 Transformation .......................................... 13  
3.3 Mini-ROCKET ............................................... 14  
3.3.1 Kernels .................................................... 14  
3.3.2 Transformation .......................................... 14  
3.4 Model ....................................................... 14  
3.4.1 Baseline - SVM .......................................... 15  
3.5 Data Collection ............................................ 15  
3.5.1 Data Annotation ........................................ 16  
3.5.2 Target Population ....................................... 17  
3.6 Data Preparation .......................................... 17
3.6.1 Preparation Pipeline ........................................ 17
3.6.2 Dataset Specifications ...................................... 19
3.6.3 Data balancing .................................................. 19
3.6.4 Dataset Reduction ............................................. 19

3.7 Experimental design/Planned Measurements ................. 20
3.7.1 Hardware and Software ...................................... 20
3.7.2 Test environment ............................................. 20

3.8 Assessing reliability and validity of the data collected ...... 21
3.8.1 Validity of method ............................................ 21
3.8.2 Reliability of method ......................................... 21
3.8.3 Data validity ................................................... 22
3.8.4 Reliability of data ............................................. 22

3.9 Planned Data Analysis .......................................... 22
3.9.1 Data Analysis Metrics ....................................... 22
3.9.2 Data Analysis Techniques ................................... 23

3.10 Evaluation framework ........................................ 24

4 Implementation ..................................................... 25
4.1 Feature Extraction and Training Configuration ............... 25
4.2 Hyperparameter Tuning - Tree-structured Parzen Estimator .. 26
4.3 Preliminary Pilot Testing ....................................... 26
4.3.1 Channel Groupings and Moving Averages ................ 27
4.3.2 Outlier Analysis ............................................. 27

5 Results and Analysis ............................................. 28
5.1 ResNET and Mini-ROCKET Comparison ...................... 28
5.2 Data Balancing and Reduction ................................ 29
5.3 Hyperparameter Tuning ........................................ 30
5.4 Model Baseline Comparison ................................... 30
5.5 Consecutive Heartbeats ........................................ 31
5.6 Accuracy Trends Relative to Electrode Group Distance from Center ............................................. 31
5.7 Class Distribution Trends Relative to Electrode Group Distance from Center ............................................. 32

6 Discussion ............................................................. 34
6.0.1 ResNET and Mini-ROCKET Comparison .................... 34
6.0.2 Data Balancing and Reduction ............................. 35
6.0.3 Hyperparameter Tuning ...................................... 35
6.0.4 Model Baseline Comparison ................................ 35
| 6.0.5   | Consecutive Heartbeats                           | 36 |
| 6.0.6   | Accuracy and Class Distribution Trends Relative to Electrode Group Distance from Center | 36 |

| 7       | Conclusions and Future work                     | 37 |
| 7.1     | Conclusions                                     | 37 |
| 7.2     | Limitations                                     | 38 |
| 7.3     | Future work                                     | 38 |

| References |                                                         | 39 |
List of Figures

1.1 EMG signal detection from the diaphragm using electrode array. (a) Schematic representation of the Edi catheter placement within the esophagus and stomach. Electrodes positioned in proximity to the diaphragm register the highest EMG signals. (b) Displays the raw signal captured by the electrode array, which is subsequently processed through an EMG isolation filter. Each EMG burst corresponds to a respiratory effort. Regions with elevated EMG signals are highlighted in yellow. 3

1.2 Illustration of the PQRST Complex in an Electrocardiogram (ECG) Signal 5

3.1 Architecture of the MiniROCKET Classifier. Features extracted via MiniROCKET are processed through a dense layer before passing through a softmax function. This ultimately categorizes the input into one of three classes: 0 (Too Far Up), 1 (Correct), or 2 (Too Far In). Here, n represents the number of features and is set to 9996 and X represents a node in the dense layer. 15

3.2 Representation of the annotation scheme. The left side of the figure displays the 8-channel EMG signal captured by the Edi catheter during a single breath. Areas of heightened EMG amplitude are highlighted with a yellow curve. The channel with the highest EMG amplitude is highlighted in pink, representing the electrode centered around the diaphragm. The right side illustrates the annotation; the electrode on the array with the highest EMG signal is marked as correctly positioned (marked in green), while the remaining electrodes are considered to be incorrectly positioned (marked in red). 16
3.3 Flowchart detailing the data preparation pipeline. (a) Depicts the segmentation of the 8-channel ECG signal into individual heartbeats, with the top channel corresponding to the first electrode in the insertion direction towards the stomach (b) Displays the 8-channel array of individual heartbeats, further segmented into paired groups. (c) Illustrates the labeling scheme for these groups: blue highlight (=0) indicates positioning Too Far Up, green highlight (=1) indicates Correct positioning, and red highlight (=2) suggests positioning Too Far In. (d) Demonstrates the downsampling from 2000 Hz to 400 Hz.

5.1 Accuracy plotted against distance from the electrode group centered around the diaphragm, using the Partial balancing scheme with 99% size reduction after TPE execution and a majority decision based on 5 consecutive heartbeats. Negative distances indicate an electrode group that is 'Too Far Out', a 0 distance signifies the correctly positioned group, and positive distances denote an electrode group that is 'Too Far In'.

5.2 Predicted class distributions plotted against distance from the electrode group centered around the diaphragm, using the Partial balancing scheme with 99% size reduction after TPE execution. Negative distances indicate an electrode group that is 'Too Far Out', a 0 distance signifies the correctly positioned group, and positive distances denote an electrode group that is 'Too Far In'.
# List of Tables

<table>
<thead>
<tr>
<th>Section</th>
<th>Table Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Dataset Specifications, detailing patient count, sample quantities, and label distributions.</td>
<td>19</td>
</tr>
<tr>
<td>3.2</td>
<td>Specifications of Dataset Balancing, illustrating the split, sample counts, and label distributions.</td>
<td>19</td>
</tr>
<tr>
<td>5.1</td>
<td>Performance comparison between ResNet and Mini-ROCKET based on a single data split. The top table utilizes the dataset from the original work at Getinge, while the bottom table employs the new dataset introduced in this thesis.</td>
<td>28</td>
</tr>
<tr>
<td>5.2</td>
<td>Accuracies and Training Durations for the three distinct balancing schemes: Full, Partial, and Unbalanced. The balancing scheme with the highest accuracy is highlighted in bold. Standard deviations are provided in parentheses.</td>
<td>29</td>
</tr>
<tr>
<td>5.3</td>
<td>Accuracies and training durations associated with each size reduction under the Partial balancing scheme. Size reductions are executed on a per-patient basis, where a reduction of 1 heartbeat corresponds to a random sampling of one heartbeat per group for each patient. The size reduction yielding the highest accuracy is emphasized in bold. Standard deviations are presented within parentheses.</td>
<td>29</td>
</tr>
<tr>
<td>5.4</td>
<td>Accuracies and training durations for the Partial balancing scheme with 99% size reduction following TPE execution. Standard Deviations are indicated in parentheses.</td>
<td>30</td>
</tr>
<tr>
<td>5.5</td>
<td>Final hyperparameter configuration following TPE execution.</td>
<td>30</td>
</tr>
<tr>
<td>5.6</td>
<td>Accuracies and training durations for the Partial balancing scheme with 99% size reduction following TPE execution and an SVM using the RBF kernel. Standard deviations are indicated in parentheses.</td>
<td>31</td>
</tr>
</tbody>
</table>
5.7 Accuracies of the Partial balancing scheme with 99% size reduction after TPE execution, for both a single heartbeat and a majority decision based on 5 consecutive heartbeats. Standard deviations are provided in parentheses.
## List of acronyms and abbreviations

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIMA</td>
<td>Autoregressive Integrated Moving Average</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional Neural Network</td>
</tr>
<tr>
<td>DCNN</td>
<td>Deep Convolutional Neural Network</td>
</tr>
<tr>
<td>ECG</td>
<td>Electrocardiogram</td>
</tr>
<tr>
<td>Edi</td>
<td>Electrical Activity of the Diaphragm</td>
</tr>
<tr>
<td>EMG</td>
<td>Electromyography</td>
</tr>
<tr>
<td>EST</td>
<td>Exponential Smoothing State Space</td>
</tr>
<tr>
<td>ICU</td>
<td>Intensive Care Unit</td>
</tr>
<tr>
<td>k-NN</td>
<td>k Nearest Neighbors</td>
</tr>
<tr>
<td>LSTM</td>
<td>Long Short-Term Memory</td>
</tr>
<tr>
<td>Mini-ROCKET</td>
<td>Minimally RandOm Convolutional KErnel Transform</td>
</tr>
<tr>
<td>ML</td>
<td>Machine Learning</td>
</tr>
<tr>
<td>RBF</td>
<td>Radial Basis Function</td>
</tr>
<tr>
<td>ResNet</td>
<td>Residual Network</td>
</tr>
<tr>
<td>RNN</td>
<td>Recurrent Neural Network</td>
</tr>
<tr>
<td>ROCKET</td>
<td>RandOm Convolutional KErnel Transform</td>
</tr>
<tr>
<td>SDG</td>
<td>Sustainable Development Goal</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>TPE</td>
<td>Tree-structured Parzen Estimator</td>
</tr>
<tr>
<td>TSC</td>
<td>Time-Series Classification</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

Machine Learning (ML) has revolutionized numerous fields, with biomedicine being one of the most prominent beneficiaries. In biomedicine, ML-driven image analysis has been instrumental in areas such as Alzheimer’s disease detection and cancer diagnosis. Concurrently, time-series data, exemplified by Electrocardiogram (ECG) signals, plays a pivotal role in diagnostics and patient monitoring. In recent years, Deep Convolutional Neural Networks (DCNNs) have been adopted in numerous studies for time-series data classification, notably in ECG signal analysis [1, 2]. While these studies have demonstrated encouraging outcomes in ECG classification, the inherent computational demands and large data necessities of DCNNs are a recognized limitation. These challenges have spurred the development of more efficient Convolutional Neural Network (CNN) variants, including RandOm Convolutional KErnelt Transform (ROCKET) and its successor, Minimally RandOm Convolutional KErnelt Transform (Mini-ROCKET).

This thesis aims to exploit the efficient architecture of Mini-ROCKET to investigate potential improvements in the existing methodology for ECG signal classification at Getinge. Specifically, we endeavor to train a Mini-ROCKET model on the company’s proprietary ECG patient dataset to assess whether we can enhance the efficiency of Electrical Activity of the Diaphragm (Edi) catheter position classification, as compared to the current Residual Network (ResNet)-based approach. We aim to measure improvement by computational cost, performance accuracy, and the user value of the resulting classifications as compared to previous Edi catheter positioning research at Getinge.
1.1 Background

The challenges surrounding manual Edi catheter positioning have been well-recognized by the clinicians for some time. Currently, the positioning of the Edi catheter is guided by manual interpretation of Electromyography (EMG) and ECG signals. These signals, emanating from the diaphragm and the heart, are captured by a 9-lead electrode array. This array produces an 8-channel bipolar measurement of potentials derived from the 9 electrodes, as illustrated in Figure 1.1.
Figure 1.1: EMG signal detection from the diaphragm using electrode array. (a) Schematic representation of the Edi catheter placement within the esophagus and stomach. Electrodes positioned in proximity to the diaphragm register the highest EMG signals. (b) Displays the raw signal captured by the electrode array, which is subsequently processed through an EMG isolation filter. Each EMG burst corresponds to a respiratory effort. Regions with elevated EMG signals are highlighted in yellow.

As illustrated in Figure 1.1a, minimal movement is sufficient to displace the electrodes from the diaphragm, increasing the risk of displacement due to factors such as patient movement, coughing, or other disturbances. Conversely, as demonstrated in Figure 1.1b, these signals must be continuously monitored and interpreted by proficient clinicians to ensure optimal positioning. While this particular signal appears distinct, not all EMG signals are similarly discernible. Within the Intensive Care Unit (ICU), it is not uncommon for patients to be connected to mechanical ventilators,
which operate based on signals detected by the Edi catheter. However, during the positioning phase, it’s not uncommon to encounter weak or absent EMG signals. This necessitates reliance on ECG interpretation, which, unfortunately, lacks the precision required for the fine-tuning of Edi catheter placement and is challenging to interpret.

Given the pivotal role of the Edi catheter in regulating both the magnitude and timing of breaths, as well as its significance in nutrition by ensuring correct placement within the stomach, its positioning is paramount. While the realm of automated Edi catheter positioning is a nascent research endeavor at Getinge, there have been prior studies addressing catheter position recognition [3, 4, 5]. These studies have demonstrated promising outcomes by employing DCNNs, such as ResNets, to classify catheter positions via radiographic imaging. However, radiography has inherent drawbacks: the time needed for exposure and image processing makes it unsuitable for continuous monitoring. Moreover, radiographic imaging subjects patients to radiation, posing risks during prolonged exposure. Thus, relying on radiography for continuous Edi catheter positioning and monitoring is not viable.

This motivated Getinge to investigate automated classification of the Edi catheter positioning in a previous study. The study hypothesized that the PQRST complex of the ECG signal, as depicted in Figure 1.2, could contain sufficient data for precise classification of the position of the Edi catheter in relation to the diaphragm. In this context, the initial peak, termed the P-wave, represents the propagation of an electrical impulse initiated by the sinoatrial node located in the right atrium, traversing both atria of the heart. The Q, R, S-waves then collectively delineate the ensuing ventricular contraction triggered by this impulse. The terminal peak, known as the T-wave, illustrates the relaxation of the ventricles following the cessation of the electrical impulse propagation [6]. Utilizing this ECG information could, in theory, enable real-time online classification, thereby circumventing the challenges posed by the absence of EMG signals.
The outcome of this study was a ResNet model, trained on an in-house dataset comprising multichannel time-series ECG data [7]. This model is capable of classifying the positioning of an Edi catheter as either incorrect or correct (a binary classification) based on an 8-channel electrode array and corresponding ECG signal. Although the model demonstrated high performance within the scope of the thesis, it is not without limitations. Specifically, the model incurs a high computational cost and a limited diversity in the training data. Additionally, its binary classification scheme offers limited information to the user, as it only distinguishes between correct and incorrect positioning. In light of these constraints, Getinge has requested for continued research to further refine the methodology.

1.2 Problem and Research Questions

The objective of this research is to address the central question: Can Mini-ROCKET be employed to enhance the existing methodology for Edi catheter positioning? By delineating 'effectiveness' as the model’s performance in terms of various accuracy metrics (e.g., Accuracy and F1-score) and 'efficiency' as the model’s training speed. The research specifically seeks to address these sub-questions:

• To what extent can Mini-ROCKET improve effectiveness and efficiency in comparison to the existing ResNet-based approach?
What strategies can be implemented to enrich the user value of the classified catheter positions?

To what extent can classification accuracy be improved by diversifying the training dataset?

To what degree can the effectiveness of the Mini-ROCKET model be increased while concurrently mitigating computational expense?

1.3 Purpose

The objective of this thesis aligns with the original research initiative undertaken at Getinge, which aims to enhance the positioning procedure of the Edi catheter through the application of ECG classification. This automation significantly alleviates the workload of clinicians responsible for manual positioning, while also facilitating automated monitoring. Such monitoring serves to promptly alert medical personnel in the event of catheter misplacement. Consequently, this approach not only economizes the time and effort required for training medical staff in manual Edi positioning but also minimizes the potential for errors in both the positioning and ongoing surveillance of the catheter’s location.

While the original research initiative demonstrated high accuracy in classifying catheter positions, it was accompanied by considerable computational costs during the model training/testing phase. In addition to the primary objective, this thesis also endeavors to investigate alternative techniques, such as Mini-ROCKET, to identify a more computationally efficient approach for catheter position classification. Implementing such methods could yield significant time and resource savings during both the training and testing phases of the model. Given that the existing model necessitates substantial GPU computational power for both training and testing, the adoption of a more efficient model like Mini-ROCKET could substantially reduce organizational expenditures for Getinge in terms of both hardware and time.

1.4 Goals

The specific goals of this thesis are delineated as follows:

1. Evaluate whether Mini-ROCKET can achieve performance metrics that are either superior to or comparable with the original ResNet approach, while simultaneously enhancing computational efficiency.
2. Enhance data diversity and methodological informativeness by incorporating additional patient samples and expanding the classification scheme to include three categories:
   - Too Far In,
   - Too Far Out.
   - Correct.

3. Conduct a comprehensive data analysis to identify novel strategies for augmenting both the effectiveness and efficacy of the model.

4. Employ hyperparameter optimization techniques to extract the maximum achievable performance from the model.

5. Undertake an in-depth analysis of the most optimized Mini-ROCKET model to assess its performance characteristics and limitations.

1.5 Research Methodology

This section provides an overview of the research methodology utilized in this thesis for the application, comparison, and analysis of the novel Mini-ROCKET-based approach. The research framework primarily encompasses data collection, data engineering, Mini-ROCKET implementation, and empirical evaluation, each facet of which will be further detailed in Chapter 3.

The dataset employed in this study is an in-house collection provided by Getinge, comprising raw signal data from ICU patients. This section will elucidate how this data can be systematically annotated and transformed into a format conducive to addressing the Edi catheter positioning problem. Subsequently, we will demonstrate how Mini-ROCKET can be effectively deployed to resolve the aforementioned problem, along with the metrics suitable for performance evaluation.

The anticipated outcomes of this research posit that Mini-ROCKET can feasibly supplant the existing ResNet model, achieving superior efficiency without compromising performance metrics.

1.6 Ethical and Societal Aspects

It's important to consider the ethical and societal implications that can occur when developing healthcare AI systems. In the context of an Edi
catheter position classifier, numerous ethical and societal challenges must be thoroughly addressed prior to deployment. One immediate concern and hurdle pertains to the availability of appropriate data. Given the limited patient size of Getinge’s dataset, there is a potential risk of inadequate patient representation for generalizing to the entire population. The integration of non-generalizable AI into the healthcare sector can swiftly lead to ethical and societal dilemmas.

Furthermore, a small patient dataset introduces the risk of overfitting, which can result in biased model predictions favoring the training data rather than offering general applicability to the broader population. This concern influenced our choice of Mini-ROCKET, which, being lightweight, poses a lower risk of overfitting compared to models with more complex architectures.

Taking into account all potential ethical and societal challenges, the continued automation and enhancement of healthcare through the application of AI, such as the Edi catheter classifier, can make a significant contribution to achieving the United Nations Sustainable Development Goals (SDGs), notably SDG 3, which focuses on good health and well-being [8].

1.7 Delimitations

While the original thesis did explore certain optimizations, it lacked comprehensive evaluation across multiple dataset splits, a limitation that could potentially impact the final results. It should be noted that the current study will not engage in extensive cross-validation of the original ResNet model due to computational constraints. This limitation is acknowledged as a potential factor that may result in a less exhaustive comparison between the two approaches. Consequently, the focus has been shifted towards optimizing the Mini-ROCKET model within the available computational resources.

Another constraint of this study is our exclusive reliance on Getinge’s proprietary dataset, with no consideration given to other data sources.

1.8 Structure of the thesis

Chapter 2 presents a comprehensive background section about about ROCKET and it’s successor Mini-ROCKET. Chapter 3 presents the methodology and techniques employed to address the research problem. Chapter 4 elaborates on the procedures followed for the implementation and application of Mini-ROCKET. Chapter 5 is dedicated to the presentation of research results and findings. Chapter 6 engages in the interpretation and discussion of
these findings. Finally, Chapter 7 concludes the thesis by drawing conclusions, outlining the model’s limitations, and proposing avenues for future research. References are appended at the end of the thesis.
Chapter 2

Background

This chapter introduces the domain of time-series classification and delineates the foundational concepts of Mini-ROCKET and its predecessor, ROCKET, emphasizing their roles in time-series classification.

2.1 Time-Series Classification

Time-Series Classification (TSC) is a fundamental task in many domains, including finance, engineering, and biomedicine. It involves predicting a label for a whole sequence based on the temporal patterns within that sequence. Over the years, various models have been developed and employed for TSC, each with its unique strengths and challenges.

- **Traditional Statistical Models:** Historically, models like Autoregressive Integrated Moving Average (ARIMA) and Exponential Smoothing State Space (EST) were popular for time-series forecasting [9]. However, for classification tasks, these models often require extensive feature engineering.

- **Machine Learning Models:** Decision trees, $k$ Nearest Neighbors ($k$-NN), and Support Vector Machine (SVM) have been adapted for TSC [10, 11, 12]. These models typically require the transformation of time-series data into a feature vector representation.

- **Deep Learning Models:** With the advent of deep learning, models like CNNs and Recurrent Neural Networks (RNNs) have gained prominence in TSC [13, 14, 15]. Long Short-Term Memorys (LSTMs) networks, a variant of RNNs, have been particularly effective due to their ability to capture long-term dependencies in sequences.
In the realm of biomedicine, time-series classification plays a pivotal role, especially in the analysis of physiological signals. ECG signals, which capture the electrical activity of the heart over time, are a prime example. Accurate classification of ECG signals can aid in diagnosing various cardiac anomalies, such as arrhythmias or myocardial infarctions. Deep learning models, particularly CNNs, have shown significant promise in this area, offering high accuracy rates [16, 1]. The ability of these models to automatically extract relevant features from raw ECG signals, without the need for manual feature engineering, makes them especially valuable for real-time monitoring and diagnosis in clinical settings.

2.2 ROCKET

ROCKET is a significant advancement within time-series classification and Convolutional Networks. It was first introduced in 2020 in a research paper titled ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels [17]. By transforming time series data using random convolutional kernels and aggregate features, the authors show that state-of-the-art classification accuracy can be achieved with a fraction of the computational time compared to other methods such as ResNet and ProximityForest.

2.3 Mini-ROCKET

In the proceedings of ROCKET, the Mini-ROCKET model, was introduced in 2021. Mini-ROCKET outperforms its predecessor, ROCKET, by achieving computational speeds that are 75 times faster while maintaining comparable classification accuracy. [18] This enhancement is attributed to modifications in kernel generation, which minimize randomness, and optimizations in feature transformation.
Methods

Chapter 3

The purpose of this chapter is to provide an overview of the research method used in this thesis. Section 3.1 describes the research process. Section 3.2 and 3.2 the architectures and techniques underpinning ROCKET and Mini-ROCKET, respectively. Section 3.4 elaborates on the linear classifier that incorporates Mini-ROCKET for our model, as well as the baseline for comparative analysis. Section 3.5 focuses on the data collection techniques used for this research. Section 3.6 shows the techniques employed for preparing the data for classification. Section 3.7 describes the experimental design. Section 3.8 explains the techniques used to evaluate the reliability and validity of the data collected. Section 3.9 describes the method used for the data analysis. Finally, Section 3.10 describes the framework selected to evaluate the Mini-ROCKET classifier.

3.1 Research Process

To address the research questions outlined in Section 1.2, a robust research design is imperative for a systematic and rigorous exploration of the issue. The most appropriate research approach for this thesis is experimental research. This method is particularly apt for machine learning and represents a design frequently employed to ascertain the relationship between cause and effect. It entails the manipulation of one or more variables while maintaining other variables constant, subsequently measuring their impact on a specific outcome.

The experimental research process comprises several stages, including:

1. Formulating a research question,
2. Designing the experimental plan,
3. Execution of the experiment,
4. Data analysis derived from the experiment, and
5. Interpretation of the results and conclusion drawing.

For the purposes of this study, experimental research will be employed to evaluate Mini-ROCKET under various configurations, encompassing both hyperparameters and data.

### 3.2 ROCKET

#### 3.2.1 Kernels

ROCKET generates a predetermined number of convolutional kernel, irrespective of the time series length. The generation process involves the random sampling of kernel attributes from predefined distributions, including kernel length, weight, bias, dilation, and padding. Notably, the kernel stride is consistently maintained at one. With the employment of a singular layer of kernels and the absence of kernel weight learning, the convolution computational cost is significantly reduced. Consequently, ROCKET is able to use a massive variety of kernels and keep computational costs minimal.

#### 3.2.2 Transformation

In the transformation phase, each randomly generated kernel is applied to every input time series, yielding a corresponding feature map. Subsequently, ROCKET undertakes the computation of two aggregate features from each feature map: the maximum value (global max pooling), and the proportion of positive values (or PPV). The maximum value features capture the highest response from the time series to the particular kernel, indicating areas of maximum similarity. Meanwhile, the PPV metric captures the frequency at which the time series exhibits a positive response to the kernel. For k kernels, these two aggregations collectively yield a total of 2k features for each time series. In effect, the only configurable hyperparameter in ROCKET is the number of kernels.
3.3 Mini-ROCKET

3.3.1 Kernels

Motivated through experimentation, Mini-ROCKET eliminates the need for random sampling of kernel parameters. It achieves this by maintaining fixed dilation and padding, constraining weights to two possible values, setting the length to 9, and deriving biases directly from the convolutional output. Consequently, the possible configurations of these two-valued kernels are reduced to \(2^9 = 512\) choices. Given that observed performance did not exhibit significant improvements beyond 10 000 kernels, Mini-ROCKET standardizes the number of kernels to 10 000.

3.3.2 Transformation

In the kernel transformation process, Mini-ROCKET eliminates the reliance on the maximum value feature. Instead, it leverages the fixed set of two-valued kernels with PPV through four pivotal optimizations:

1. Computing PPV for the weights and their negatives.
2. Utilizing the convolution output to derive multiple features efficiently.
3. Bypassing multiplication operations during the convolution process.
4. Processing all kernels simultaneously for each specified dilation.

3.4 Model

In our study, we re-implemented Mini-ROCKET based on the methodology presented in the original research paper [18]. Given the promising results observed with Mini-ROCKET when combined with linear classifiers in the original work, we opted to utilize Mini-ROCKET as a feature extractor. The extracted features are then input into a linear classifier, specifically a dense layer, which is subsequently followed by a softmax function, as illustrated in Figure 3.1.
Figure 3.1: Architecture of the MiniROCKET Classifier. Features extracted via MiniROCKET are processed through a dense layer before passing through a softmax function. This ultimately categorizes the input into one of three classes: 0 (Too Far Up), 1 (Correct), or 2 (Too Far In). Here, n represents the number of features and is set to 9996 and X represents a node in the dense layer.

While the default feature set of Mini-ROCKET consists of 10,000 features, due to the architectural design, only the nearest multiple of 84 among these features are extracted.

3.4.1 Baseline - SVM

For the baseline model, SVM, we employed Mini-ROCKET as a feature extractor, similar to the approach illustrated in Figure 3.1. However, instead of using a dense and softmax layer, we incorporated an SVM with the Radial Basis Function (RBF) kernel to predict one of the three classes.

3.5 Data Collection

All experiments were executed using Getinge’s proprietary dataset, which comprises data from ICU patients. This dataset encompasses ECG and EMG data from 87 patients, acquired via the Edi catheter, as previously depicted in Figure 1.1. The EMG data played a crucial role during the annotation phase, whereas the ECG signal was pivotal for position inference. Subsequent sections will detail the techniques used for annotation and describe the target population for the dataset.
3.5.1 Data Annotation

The annotation process utilized the EMG signals from the dataset. Given that each patient maintained a stationary position of the Edi catheter, only one annotation per patient was made. The annotation scheme is illustrated in Figure 3.2, where the channel with the highest EMG amplitude is annotated as correct.

![8-channel EMG signal](image)

Figure 3.2: Representation of the annotation scheme. The left side of the figure displays the 8-channel EMG signal captured by the Edi catheter during a single breath. Areas of heightened EMG amplitude are highlighted with a yellow curve. The channel with the highest EMG amplitude is highlighted in pink, representing the electrode centered around the diaphragm. The right side illustrates the annotation; the electrode on the array with the highest EMG signal is marked as correctly positioned (marked in green), while the remaining electrodes are considered to be incorrectly positioned (marked in red).
As most signal recordings from these patients were taken post-positioning, the majority of the signals appear to be correctly centered around the diaphragm. Notably, either lead 4 or 5 on the electrode array exhibited the highest EMG amplitudes in most of the recordings.

3.5.2 Target Population

All participants in the study were adults, thus defining the target population. The applicability of this data to pediatric populations remains an avenue for future research.

3.6 Data Preparation

In line with previous study, we adopted a comparable data preparation pipeline. However, unlike prior work, we eliminated heartbeat averaging and introduced pairwise channel grouping in place of 3-way channel grouping. The rationale behind these modifications is elaborated upon in the preliminary pilot testing, as discussed in Section 4.3. To address the research question, "What strategies can enhance the informativeness of classified catheter positions?", from Section 1.2, we introduced an additional class to specify whether the inserted Edi catheter should be advanced further, retracted, or remain stationary. This provides clinicians with added insight regarding the subsequent positioning action, whether to push it in further or pull it out. Subsequent sections will elaborate on the revised data preparation pipeline utilized to convert the ECG data into a format suitable for training our classifier. We will then describe the resultant dataset and conclude by discussing various balancing techniques applied to this dataset.

3.6.1 Preparation Pipeline

The data preparation pipeline is illustrated in Figure 3.3 and comprises the following four steps:

a **Heartbeat segmentation.** Utilizing a given R-peak detection algorithm supplemented with manual adjustments, we sampled 300 milliseconds both preceding and succeeding each R-wave peak. This approach ensures capture of the entire PQRST complex within the ECG signal, as previously shown in Figure 1.2. Consequently, each 8-channel ECG signal for every patient was segmented into discrete heartbeats.
Group segmentation. To capture channel dependencies, we transformed the data into pairwise channel groups, where the last channel of one pair serves as the first channel of the subsequent pair. This approach effectively converts the classification task into a dual-channel time series classification problem.

c Group labeling. In this phase, we systematically generate labels for each group on a per-patient basis. As illustrated in Figure 3.3, if a group contains the channel denoted as correctly centered around the diaphragm, that group is labeled as "Correct". Groups above are labeled as "Too Far Up", and groups below are labeled as "Too Far In".

d Down sampling. Each signal is downsampled from its original sampling rate of 2000 Hz to 400 Hz. This reduction is implemented to decrease the size of the input data, thereby enhancing training speeds. A rate of 400 Hz was selected as a safe upper bound, as frequencies below this
threshold have been demonstrated to capture the critical information of the PQRST complex within the ECG signal [19].

### 3.6.2 Dataset Specifications

Table 3.1 presents the dataset derived from the data preparation pipeline detailed in Section 3.6.1. This dataset was employed for all experiments.

<table>
<thead>
<tr>
<th>No. Patients</th>
<th>No. Samples</th>
<th>Too Far Up = 0</th>
<th>Correct = 1</th>
<th>Too Far In = 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>87</td>
<td>492 226</td>
<td>195 214</td>
<td>138 001</td>
<td>159 011</td>
</tr>
</tbody>
</table>

Table 3.1: Dataset Specifications, detailing patient count, sample quantities, and label distributions.

We utilized training, validation, and test splits, incorporating 10-fold cross-validation to enhance result robustness. Both the validation and test sets were kept disjoint from training, operating under the premise that patients would be unknown during inference. The data was divided into an 80/10/10 split for training, validation, and testing, respectively.

### 3.6.3 Data balancing

We applied two balancing techniques to the dataset: Partial and Full. In the Partial technique, the largest class (‘Too Far Up’) was downsampled to match the second largest class (‘Too Far In’). For the Full technique, both the ’Too Far Up’ and ‘Too Far In’ classes were downsampled to match the size of the smallest class, ‘Correct’, thereby achieving a fully balanced dataset. The specifications of these splits are detailed in Table 3.2.

<table>
<thead>
<tr>
<th>Split</th>
<th>No. Samples</th>
<th>Too Far Up = 0</th>
<th>Correct = 1</th>
<th>Too Far In = 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Partial</td>
<td>456 023</td>
<td>159 011</td>
<td>138 001</td>
<td>159 011</td>
</tr>
<tr>
<td>Full</td>
<td>414 003</td>
<td>138 001</td>
<td>138 001</td>
<td>138 001</td>
</tr>
</tbody>
</table>

Table 3.2: Specifications of Dataset Balancing, illustrating the split, sample counts, and label distributions.

### 3.6.4 Dataset Reduction

Upon examining the data, we observed patterns in the heartbeats of each patient. Heartbeats within the same recording exhibited strong similarities. Notably, heartbeats at the end of a recording displayed slight difference
compared to those at the beginning. However, consecutive heartbeats in the
timeline were strikingly alike. Based on this observation, we implemented
a heartbeat sample reduction for each patient, decreasing the samples from
each recording by 50%, 90%, and 99%, extending to a single heartbeat
per patient. This approach was motivated by the hypothesis that, given the
observed similarity in heartbeats, it might be possible to enhance processing
speed without compromising accuracy.

3.7 Experimental design and
Planned Measurements

3.7.1 Hardware and Software

All experiments were conducted on a MacBook Pro 2023 laptop. The
specifications are detailed below:

- **Central Processing Unit (CPU):** Apple M2 Pro
- **Random Access Memory (RAM):** 32 GB
- **Operating System (OS):** Ventura 13.3

In addition, MATLAB R2023a was utilized for data sampling and
preparation, and Python 3.11 was employed for data balancing and model
experimentation.

3.7.2 Test environment

All testing was conducted in Jupyter notebook. To execute the notebook, the
following dependencies are needed:

- NumPy
- Pandas
- Pickle
- Torch
- Datetime
- MiniROCKET multivariate
• Sklearn.metrics
• Matplotlib.pyplot
• Time
• Os
• IPython.display
• Copy

### 3.8 Assessing reliability and validity of the data collected

This section delineates the measures implemented to uphold the validity and reliability of the methodology and the dataset.

#### 3.8.1 Validity of method

To validate our method, we will juxtapose its accuracy with the previously established ResNet model, which has already demonstrated its suitability for the Edi catheter positioning problem. Equivalent or superior performance would signify not just applicability, but also enhanced efficacy. Additionally, we will benchmark against an SVM baseline, given its well-researched nature and established use as a comparative standard. We will employ three distinct accuracy metrics: prediction accuracy, weighted F1-score, and unweighted F1-score, elaborated further in Section 3.9.1. These metrics collectively enhance the evaluation’s robustness and offer diverse insights into the method’s effectiveness.

#### 3.8.2 Reliability of method

The method’s reliability is reinforced through the use of 10-fold cross-validation, ensuring that folds are consistently distinct from both training and validation data. This approach mitigates the influence of chance on results, bolstering confidence in the method’s consistency. Furthermore, we present the standard deviation across all folds, capturing any potential performance disparities between splits.
3.8.3 Data validity

The validity of the data is bolstered by its origin from real-world settings. All the patients in the dataset volunteered while situated in an actual ICU environment, ensuring the results are representative and can be generalized to real-life scenarios. Given that the test splits remain distinct from both validation and training, we further emulate genuine conditions, enhancing the data’s validity.

3.8.4 Reliability of data

While the data annotation was performed manually by the author, introducing a potential source of unreliability, it was subsequently reviewed for accuracy by experts at Getinge. This two-step process ensures a higher degree of reliability in the annotations. Moreover, the dataset was rigorously inspected for any inconsistencies or unreliable patient data. Specific outliers, such as missing EMG signals, were identified and excluded. Any anomalous segments within the ECG signal recordings were also removed, ensuring optimal data integrity.

3.9 Planned Data Analysis

The data analysis comprises two main components. Initially, we discuss the metrics adopted in this study, followed by a review of the data analysis techniques utilized.

3.9.1 Data Analysis Metrics

This study employs three distinct metrics for performance evaluation: Accuracy, Weighted Average (WA) F1-Score, and Unweighted Average (UA) F1-Score. Accuracy is defined as the proportion of correct predictions relative to the total number of predictions as follow:

$$\text{Accuracy} = \frac{\text{Number of correct prediction}}{\text{Number of total predictions}}$$

Accuracy serves as a pertinent metric when datasets are balanced, meaning an even distribution exists among the classes. Consider a scenario where the dataset comprises 99% of one class and only 1% of another. In such a situation, a model might default to predicting the dominant class consistently, achieving a 99% accuracy. However, in the presence of class imbalance, the
F1-score offers a more robust measure of performance. Prior to delving into the mechanics of the F1-score, it’s essential to define its foundational elements: Precision and Recall. Precision is defined as:

\[
\text{Precision} = \frac{\text{True positives}}{\text{True positive} + \text{False positives}}
\]

The precision metric addresses the question: Of all positive identifications, what proportion was accurate? Precision becomes particularly valuable when false positives bear significant consequences. Conversely, recall is defined as:

\[
\text{Recall} = \frac{\text{True positives}}{\text{True positive} + \text{False negative}}
\]

Recall addresses the question: Of all actual positives, what proportion was correctly identified? As a result, the F1-score is defined as:

\[
F1\text{-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\]

F1-score combines both precision and recall into a single metric. F1-Score provides a harmonized mean of precision and recall, with the WA considering the importance or weight of each class, whereas the UA treats all classes equally. These metrics have been chosen as they offer a comprehensive view of the model’s performance across diverse scenarios and are well-accepted in the research community.

Additionally, we measure training time as an indicator of the models’ computational efficiency.

### 3.9.2 Data Analysis Techniques

While the aforementioned metrics provide insight into the accuracy of the method, they may not offer a comprehensive assessment, as they implicitly treat all electrode groupings as equally crucial. A more nuanced accuracy measure would factor in the distance of the electrode group from the diaphragm’s center. For instance, consider a patient whose central group, as derived from Section 3.6, is at position 4. Minor shifts in the Edi catheter, say by one group, might not have significant implications, given its proximity to the diaphragm’s center. In contrast, larger deviations, positioning the catheter toward the extremities, could pose greater concerns. Hence, we propose an analytical approach that plots the accuracy of each group in relation to its
distance from the diaphragm-centered group.

Augmenting this approach, another perspective involves charting the predicted class distribution based on distance from the group centered around the diaphragm. This visualization offers enhanced understanding of the resultant inferences. For instance, if we observe a position a few units above the central group, and the classification for 'Too Far Out' dominates at 90%, with 'Correct' constituting the remaining 10%, such a distribution can be interpreted with heightened confidence, especially given the absence of conflicting classifications in the other direction.

Lastly, considering the time-series structure of the data, we propose employing a majority decision mechanism over a set of consecutive heartbeats, such as five. This approach may serve to reduce ambiguity in instances where the model exhibits uncertainty between two classes.

### 3.10 Evaluation framework

This thesis aims to apply the Mini-ROCKET classifier (refer to Chapter 4) to the Edi catheter positioning problem, utilizing the dataset delineated in Section 3.6.2. The implementation will be deemed successful if it achieves accuracy comparable to the previous ResNet approach (discussed in Section 1.1) but surpasses it in terms of computational speed. By introducing the three additional classes, as detailed in Section 3.6, the classifier not only becomes more informative regarding the catheter’s position but also benefits from a more diverse training dataset due to the increased patient count. Additionally, to mitigate computational costs, we aim to experiment with dataset size reduction. We further intend to optimize the classifier by engaging in automatic hyperparameter tuning, as elaborated in Section 4.2. Finally, the performance of our Mini-ROCKET classifier will be benchmarked against a baseline SVM. All experimental analyses will employ the performance metrics outlined in Section 3.9.1, and the model’s ultimate evaluation will adhere to the methods described in Section 3.9.2.
Chapter 4

Implementation

The practical implementation of our research methodology necessitated a series of intricate steps to ensure optimal performance, especially given the architectural nuances of Mini-ROCKET. This section elucidates the procedures adopted for feature extraction, training configurations, hyperparameter tuning, and preliminary pilot testing. We address the challenges posed by large datasets, introduce a caching-based feature extraction method tailored for Mini-ROCKET, and detail the hyperparameters chosen for training. Furthermore, we delve into the Tree-structured Parzen Estimator (TPE) approach for hyperparameter optimization and discuss preliminary tests conducted, including channel groupings, moving averages, and outlier analysis. These foundational steps paved the way for the subsequent results and analyses presented in this study.

4.1 Feature Extraction and Training Configuration

Given the architectural constraints of Mini-ROCKET, it does not perform optimally with large datasets, such as the patient dataset utilized in this study. To address this, we devised a caching-based feature extraction method. Specifically, we iterated over the dataset in chunks of size 4096, extracted features, cached them, and subsequently used the cache in the training process.

We trained our classifier utilizing PyTorch and employed stochastic gradient descent with mini-batches. The training was executed using the following hyperparameter configuration:

- **Loss function**: CrossEntropyLoss
- **Optimizer**: Adam Algorithm
- **scheduler**: ReduceLROnPlateau
- **Batch size**: 256
- **Learning rate**: 0.0001
- **Patience**: 5

### 4.2 Hyperparameter Tuning - Tree-structured Parzen Estimator

For hyperparameter optimization, we utilized the **TPE**, a widely recognized method for automatic hyperparameter tuning. **TPE** utilizes hyperparameter search space distributions in the form of non-parametric densities to find the optimal hyperparameters. **TPE** can be represented as \( p(x|y) \) using two distinct densities:

\[
p(x|y) = \begin{cases} 
\ell(x), & \text{if } y < y^* \\
g(x), & \text{if } y \geq y^*
\end{cases}
\]

where \( \ell(x) \) is the density formed by using the observation in the non-parametric densities \( x \) such that corresponding loss \( y \) was less than \( y^* \) and \( g(x) \) is the density formed by using the remaining observations. The **TPE** algorithm chooses \( y^* \) to be some quantile \( \gamma \) of the observed \( y \) values, so that \( p(y < y^*) = \gamma \). And by maintaining sorted lists of observed variables in \( O = (x, y) \), the runtime of each iteration of the **TPE** algorithm can scale linearly in \(|O|\) and linearly in the number of variables (dimensions) being optimized.

### 4.3 Preliminary Pilot Testing

Before delving into the main results and their analyses, we conducted a series of preliminary tests. These were not included in the results section, as they did not significantly enhance performance. The key tests are detailed in the following subsections.
4.3.1 Channel Groupings and Moving Averages

The original ResNet study utilized channel groupings of size 3, as opposed to the size 2 discussed in Section 3.6. Additionally, they incorporated a moving average across heartbeats. While this strategy was tested in our preliminary phase, it did not yield a notable improvement in performance. Hence, we settled on groupings of size 2, which consistently demonstrated marginally better performance than the three-channel groupings with and without moving averages.

4.3.2 Outlier Analysis

We also conducted a preliminary test focused on the analysis of ECG outliers. Specifically, we considered patients with atypical ECG signals (e.g., varying amplitudes, higher heart rate) that were associated with reduced accuracy. However, as we expanded the dataset from the 28 patients in the original ResNet study to 87 patients for our research, the removal of these outliers didn’t significantly boost performance. We opted to retain these data points in our dataset, hypothesizing that they might serve as beneficial regularizers.
Chapter 5
Results and Analysis

In this chapter, we present the results from the different experiments performed on the Mini-ROCKET classifier.

5.1 ResNET and Mini-ROCKET Comparison

The dataset described in Section 3.6.2, in conjunction with the dataset from the original work at Getinge [7], was utilized to implement the prior research methodology using both ResNet and our Mini-ROCKET classifier. The corresponding accuracy and training durations are presented in Table 5.1. The results indicate comparable performance; however, the method demonstrates significantly enhanced efficiency, with approximately a 67-fold reduction in training time.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Training Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet</td>
<td>73.4%</td>
<td>93 min</td>
</tr>
<tr>
<td>MiniRocket</td>
<td>73.2%</td>
<td>1 min 9 secs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Training Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ResNet</td>
<td>56.9%</td>
<td>143 min</td>
</tr>
<tr>
<td>MiniRocket</td>
<td>57.8%</td>
<td>2 min 9 secs</td>
</tr>
</tbody>
</table>

Table 5.1: Performance comparison between ResNet and Mini-ROCKET based on a single data split. The top table utilizes the dataset from the original work at Getinge, while the bottom table employs the new dataset introduced in this thesis.
5.2 Data Balancing and Reduction

The outcomes from the 10-fold cross-validation using various balancing techniques detailed in Section 3.6.3 are showcased in Table 5.2. It is evident that the partial balancing approach yields superior performance, as it achieves the highest accuracy across all metrics and exhibits a negligible difference in training duration compared to both the full and unbalanced methods. Consequently, we opted for the partial scheme in all subsequent experiments.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Accuracy</th>
<th>F1-score (WA)</th>
<th>F1-score (UA)</th>
<th>Training Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>56.3 (±8.8)</td>
<td>55.9 (±8.9)</td>
<td>54.8 (±9.5)</td>
<td>1 min 53 secs</td>
</tr>
<tr>
<td><strong>Partial</strong></td>
<td><strong>61.0 (±6.8)</strong></td>
<td><strong>61.0 (±6.7)</strong></td>
<td><strong>60.6 (±6.9)</strong></td>
<td><strong>2 min</strong></td>
</tr>
<tr>
<td>Unbalanced</td>
<td>57.8 (±4.2)</td>
<td>57.8 (±4.5)</td>
<td>56.2 (±5.1)</td>
<td>2 min 9 secs</td>
</tr>
</tbody>
</table>

Table 5.2: Accuracies and Training Durations for the three distinct balancing schemes: Full, Partial, and Unbalanced. The balancing scheme with the highest accuracy is highlighted in bold. Standard deviations are provided in parentheses.

The outcomes of the data reduction, as detailed in Section 3.6.4, are showcased in Table 5.3. An enhancement in performance across all metrics is evident for each reduction level, with the 99% reduction emerging as the most optimal. Consequently, we selected the 99% reduction for all following experiments.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>F1-score (WA)</th>
<th>F1-score (UA)</th>
<th>Size Reduction</th>
<th>Training time</th>
</tr>
</thead>
<tbody>
<tr>
<td>64.9 (±4.8)</td>
<td>65.0 (±4.8)</td>
<td>64.0 (±4.5)</td>
<td>50%</td>
<td>1 min 4 sec</td>
</tr>
<tr>
<td>70.3 (±5.7)</td>
<td>70.2 (±5.7)</td>
<td>69.6 (±5.7)</td>
<td>90%</td>
<td>24 secs</td>
</tr>
<tr>
<td>71.7 (±3.1)</td>
<td>71.7 (±3.0)</td>
<td>70.6 (±3.3)</td>
<td>99%</td>
<td>14 secs</td>
</tr>
<tr>
<td>66.8 (±6.3)</td>
<td>66.9 (±5.9)</td>
<td>65.9 (±6.6)</td>
<td>1 heartbeat</td>
<td>12 secs</td>
</tr>
</tbody>
</table>

Table 5.3: Accuracies and training durations associated with each size reduction under the Partial balancing scheme. Size reductions are executed on a per-patient basis, where a reduction of 1 heartbeat corresponds to a random sampling of one heartbeat per group for each patient. The size reduction yielding the highest accuracy is emphasized in bold. Standard deviations are presented within parentheses.
5.3 Hyperparameter Tuning

We employed automated hyperparameter tuning utilizing TPE, as described in Section 4.2. The advantages in performance are delineated in Table 5.4, while the resultant hyperparameters are presented in Table 5.5. In comparison to the prior section, there is a slight improvement, approximately a 1% increase in performance.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>F1-score (WA)</th>
<th>F1-score (UA)</th>
<th>Training Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>72.7 (±3.8)</td>
<td>72.5 (±3.8)</td>
<td>71.5 (±4.1)</td>
<td>16 secs</td>
</tr>
</tbody>
</table>

Table 5.4: Accuracies and training durations for the Partial balancing scheme with 99% size reduction following TPE execution. Standard Deviations are indicated in parentheses.

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learning rate</td>
<td>$1 \times 10^{-4}$</td>
</tr>
<tr>
<td>Patience</td>
<td>3</td>
</tr>
<tr>
<td>Mini-Batch size</td>
<td>115</td>
</tr>
</tbody>
</table>

Table 5.5: Final hyperparameter configuration following TPE execution.

5.4 Model Baseline Comparison

The outcomes of the model baseline comparison with an SVM, as detailed in Section 3.4.1 are showcased in Table 5.6. The results clearly indicate that the Dense trained linear classifier employing Mini-ROCKET surpasses the performance of Mini-ROCKET integrated with an SVM. Although the accuracy and F1-scores are relatively close, the training duration for the model incorporating a dense layer is markedly more efficient.
### 5.5 Consecutive Heartbeats

The outcomes of the accuracy measure, as detailed in the latter portion of Section 3.9.2, are presented in Table 5.7.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>F1-score (WA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense</td>
<td>SVM</td>
</tr>
<tr>
<td>72.7 (±3.8)</td>
<td>69.8 (±3.5)</td>
</tr>
</tbody>
</table>

Table 5.7: Accuracies of the Partial balancing scheme with 99% size reduction after TPE execution, for both a single heartbeat and a majority decision based on 5 consecutive heartbeats. Standard deviations are provided in parentheses.

### 5.6 Accuracy Trends Relative to Electrode Group Distance from Center

The outcomes of the data analysis technique detailed in first part of Section 3.9.2 are presented in Figure 5.1. As depicted in the figure, both accuracy and F1-scores increase when classifying groups situated further from the central group. Accuracies and F1-scores range from 80% and approach 100% as the distance from the center increases.
Figure 5.1: Accuracy plotted against distance from the electrode group centered around the diaphragm, using the Partial balancing scheme with 99% size reduction after TPE execution and a majority decision based on 5 consecutive heartbeats. Negative distances indicate an electrode group that is 'Too Far Out', a 0 distance signifies the correctly positioned group, and positive distances denote an electrode group that is 'Too Far In'.

5.7 Class Distribution Trends Relative to Electrode Group Distance from Center

The outcomes from the data analysis technique detailed in second part of Section 3.9.2 are presented in Figure 5.2. As illustrated in the figure, positive distances display a realistic distribution spread where errors predominantly align with the class closer in distance. Conversely, for negative distances, there appears to be a more ambiguous error distribution, as both false classes exhibit similar distributions irrespective of their distance from the true class.
Figure 5.2: Predicted class distributions plotted against distance from the electrode group centered around the diaphragm, using the Partial balancing scheme with 99% size reduction after TPE execution. Negative distances indicate an electrode group that is ‘Too Far Out’, a 0 distance signifies the correctly positioned group, and positive distances denote an electrode group that is ‘Too Far In’.
Chapter 6
Discussion

This chapter delves into and discussed the findings presented in Chapter 5. The discussion will mirror the structure of the results. Within this chapter, our aim is to elucidate the significance and the underlying reasoning of the observed outcomes.

6.0.1 ResNET and Mini-ROCKET Comparison

With a comparable accuracy and a remarkable 67-fold reduction in training time, Mini-ROCKET clearly stands out as the more efficient method for Edi catheter position classification. This significant reduction in training duration facilitates efficient model analysis across multiple k-folds, even on a CPU. This efficiency presents substantial value for Getinge, both in terms of cost and time. There’s no pressing need for substantial investments in GPUs for model training and testing, nor is there a need to allocate extended working hours awaiting results.

On the other hand, the accuracy metrics of approximately 57% for ResNet and 58% for Mini-ROCKET indicate a decrease from the results presented in the prior thesis [7]. This divergence can be attributed to several factors. When evaluating both the ResNet and Mini-ROCKET models using the dataset from the previous thesis work at Getinge, we could not replicate the 88% accuracy of the original study, achieving a peak of around 73% as shown in Table 5.1. This might suggest that the original study benefited from a particularly favorable data split, possibly training and testing on less complex patient data. Another potential factor is the introduction of a third class. Moving from a binary 50/50 prediction to a ternary 33/33/33 prediction inherently increases classification complexity, which might account for the observed dip in performance.
6.0.2 Data Balancing and Reduction

The exploration of different balancing techniques, as presented in Table 5.2, underscores the efficacy of the partial balancing approach. Notably, this method not only outperforms in terms of accuracy but also maintains a competitive training time, aligning closely with the full and unbalanced techniques.

Shifting focus to data reduction, insights from Table 5.3 reveal performance improvements at distinct reduction levels. Notably, a 99% data size reduction emerges as the most favorable, balancing accuracy with computational efficiency. This significant reduction, without sacrificing the model’s predictive prowess, is noteworthy. This suggests that Getinge might benefit more from increasing the number of patients rather than securing extended recordings for each individual. The pronounced accuracy boost at 99% can be traced back to the inherent similarity of heartbeats within a single recording. Only heartbeats spaced further apart in recordings exhibit noticeable differences. This might elucidate why a 99% random sampled reduction, equating to approximately 5-10 heartbeats depending on the recording length, outperforms a single random sampled heartbeat per file. The 99% reduction provides a diverse set of heartbeats for training, accommodating the subtle variations in heartbeats spaced apart in recordings. This creates a more normalized training dataset, capturing nuances that a single random heartbeat might miss.

The capacity to uphold high accuracy amidst such pronounced data reduction further amplifies the time and cost benefits realized for Getinge.

6.0.3 Hyperparameter Tuning

The automated hyperparameter tuning yielded only marginal improvements in performance, as evidenced by a minimal increase in accuracy presented in Table 5.4. Nonetheless, this can be interpreted as a favorable outcome, underscoring the model’s robustness given its insensitivity to specific parameter variations. Such a minimal enhancement may also suggest that we are approaching the model’s performance ceiling.

6.0.4 Model Baseline Comparison

Having evaluated the Mini-ROCKET classifier against both ResNet and an SVM, it is evident that Mini-ROCKET paired with a dense layer exhibits strong performance. In the SVM baseline experiment, we observed a modest increase
in both Accuracy and F1-score, with a significant speed advantage favoring Mini-ROCKET. These findings are consistent with prior research [18, 20], which highlighted the robust performance of Mini-ROCKET when paired with a linear classifier.

6.0.5 Consecutive Heartbeats

Table 5.7 clearly indicates that employing a majority decision based on 5 consecutive heartbeats enhances accuracy metrics. While there’s a modest improvement in accuracy, the Weighted F1-score sees a more pronounced increase. This can likely be attributed to the minor class imbalance, with the F1-score (WA) offering a more comprehensive measure that captures this nuance. In the context of real-time classification, this approach offers practicality. A single heartbeat can be classified instantaneously, and as more data accumulates, the majority decision mechanism, utilizing 5 or more heartbeats, can be invoked to further refine accuracy during positioning.

6.0.6 Accuracy and Class Distribution Trends Relative to Electrode Group Distance from Center

Figure 5.1 The accuracy ranges from 80% (even higher for F1-score) to 100% as we move further from the center. As delineated in Section 3.9.2, these peripheral regions are particularly crucial for accuracy.

Additionally, Figure 5.2 offers insights into the error distribution across predicted classes. For instances categorized as 'Too Far In', errors predominantly lean towards the 'Correct' class, which is intuitively consistent given its proximity compared to the 'Too Far Out' class. This pattern is also discernible at the center. However, when moving 'Too Far Out', the model’s certainty wanes, leading to a more distributed error across the remaining classes.
Chapter 7

Conclusions and Future work

In this chapter, we will delineate conclusions pertaining to the model in Section 7.1. Subsequently, Section 7.2 will address the limitations associated with the models. Finally, Section 7.3 will explore potential directions for future research.

7.1 Conclusions

In this study, we addressed the limitations of speed and data inherent to DCNNs by leveraging the efficient Mini-ROCKET architecture, combined with a linear classifier and optimized using TPE. This model demonstrated comparable effectiveness, as gauged by Accuracy and F1-score metrics, to the prior ResNet approach employed at Getinge. Notably, it surpassed ResNet in efficiency, achieving a training speed that was approximately 540 times faster.

Additionally, we enhanced user utility by introducing a third class, providing directional guidance for Edi catheter positioning. This contrasts with the original research which only indicated correctness. By incorporating data from an additional 60 patients, we aimed to bolster the model’s generalization capabilities. Despite this, the static Accuracy measures suggest we might be nearing the model’s performance ceiling.

To further enhance the model’s effectiveness without incurring computational costs, we implemented a 99% data size reduction, which not only improved training times but also further improved classification accuracy. This underscored the importance of patient diversity over recording length. Automated hyperparameter tuning using the TPE algorithm not only yielded a modest performance boost without compromising efficiency but also provided a streamlined approach to identifying the optimal hyperparameters,
eliminating the need for extensive experimentation. Moreover, the majority decision approach, based on five consecutive heartbeats, elevated the model’s accuracy to 75% and its F1-score to 79%.

A deeper dive into the data revealed that conventional accuracy metrics might not fully capture the model’s performance nuances. By plotting accuracy against distance from the diaphragm-centered group, we discerned that both accuracy and F1-score metrics improved as the distance increased. Given the criticality of accurate positioning, especially when the Edi catheter is significantly misaligned, these distant regions should be weighted more heavily in evaluations. This adjustment suggests the model’s true performance lies between 80% and 100% for both accuracy and F1-score.

In summary, this research addressed all posed questions, delivering tangible value to Getinge in terms of cost and time savings. The need for Getinge to invest in high-end GPU equipment is obviated, and CPU-based training and testing become more time-efficient. The model’s directional guidance enhances user experience for Getinge’s clientele. By considering the distance from the diaphragm as a performance metric, we have presented a model that excels in accuracy, speed, and compactness.

### 7.2 Limitations

A prominent limitation of this study is the constrained performance when centered around the diaphragm. While the inclusion of 60 new patients represents a substantial increase compared to the previous thesis work at Getinge, it remains uncertain whether this sample size sufficiently generalizes to the entire adult population. This question extends beyond the purview of this thesis. Additionally, the dataset featured a limited number of patients with incorrect positioning, which could potentially influence the results.

### 7.3 Future work

While the dataset comprises 87 patients, it remains uncertain whether this can accurately generalize to the entire adult population. To assess its generalization capabilities, it would be beneficial to deploy the model in real-time and evaluate its performance, potentially even extending the testing to pediatric patients. As we transition to an online version, integrating a mechanism to aggregate results from all seven electrode groups to classify the entire array’s position would be advantageous. Given the scarcity of
data for incorrectly positioned Edi catheters, a future research direction could focus on employing and testing data from extreme positions. Considering our method’s limitations in classifying the ‘Correct’ class and its heightened accuracy in identifying extreme misplacements, exploring techniques for finer catheter positioning adjustments could present a valuable avenue for subsequent studies.
References


