An Embedding Multitask Neural Network for Efficient Arrhythmia Detection

Guilherme Silva[1], Arthur Negrão[1], Gladston Moreira[1], Eduardo Luz[1], Pedro Silva[1]

[1]Departament of Computing (DECOM) – Federal University of Ouro Preto (UFOP)
Ouro Preto – MG – Brasil

{guilherme.lopes,arthur.negrao}@aluno.ufop.edu.br

{gladston,eduluz,silvap}@ufop.edu.br

Abstract. *This study addresses the critical need for prompt detection of life-threatening ventricular arrhythmias. We explore the application of neural networks within the constraints of Implantable Cardioverter Defibrillators to improve early arrhythmia detection. Our proposed neural network methodology leverages multitask learning, aiming to enhance detection efficiency by concurrently learning to identify ventricular arrhythmias and estimate RR intervals from intracardiac electrograms. Implemented on the NUCLEO-L432KC board, with limited memory and processing capacity, our approach achieved an F_β score of 0.88, with a low inference latency of 59.96 ms. These results demonstrate the feasibility of integrating advanced neural network capabilities within Implantable Cardioverter Defibrillators (ICDs).*

1. Introduction

Ventricular tachycardia (VT) and ventricular fibrillation (VF) stand as critical indicators in the context of Sudden Cardiac Death (SCD), with VF being particularly feared due to its rapid progression and severe implications. These arrhythmias account for more than 60% of out-of-hospital cardiovascular fatalities [Physiopedia 2022], with VF precipitating immediate disruptions in cardiac electrical signaling, leading to uncoordinated heart muscle contractions and damaged blood circulation. The urgency for prompt rhythm correction is paramount, highlighting the critical need for rapid and precise detection of such arrhythmic events [Myerburg et al. 1992, Zipes and Wellens 1998].

Implantable Cardioverter Defibrillators (ICDs) are sophisticated devices surgically inserted beneath the skin (subcutaneously) that continuously monitor cardiac activity to identify ventricular tachyarrhythmias and administer electrical countershocks to revert the heart to its normal rhythm when needed [Mirowski 1985]. ICDs have become indispensable in the management of patients at high risk for Sudden Cardiac Death (SCD), providing a critical intervention to prevent mortality [DiMarco 2003]. Timely detection and intervention are crucial for enhancing survival prospects, therefore specialized algorithms are paramount [Zanker et al. 2016, Madhavan and Friedman 2013].

In recent years, neural networks have emerged as highly effective tools for identifying complex patterns in diverse domains, including those in medical and physiological research [Mousavi and Afghah 2019]. Inspired by the vast literature on automatic arrhythmia classification using electrocardiogram (ECG) data [Luz et al. 2016], this study posits a hypothesis: neural networks can be efficiently implemented on hardware suitable for integration into ICDs, yielding performance that rivals traditional methodologies and, more importantly, on time. To explore this hypothesis, we propose a neural network methodology to operate within the resource constraints of an NUCLEO-L432KC [STMicroelectronics 2020]. This microcontroller board, characterized by its 256KB Flash memory, 64KB SRAM, and 80 MHz CPU, is suitable for computational tasks in ICDs as presented by [Jia et al. 2023]. Figure 1 presents the STM32L432 processor, measuring a 5.30 mm by 5.30 mm.

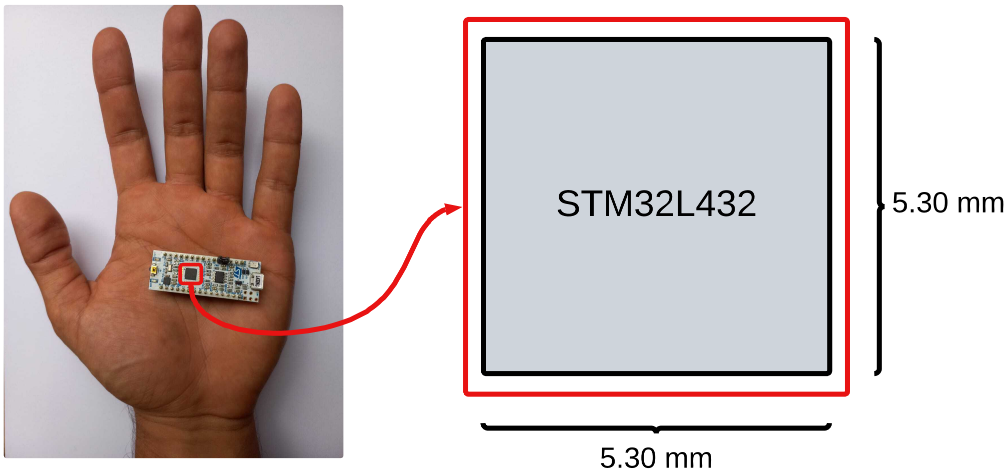

Figure 1. The NUCLEO-L432KC board against a human hand for scale, emphasizes the remarkably compact size of STM32L432 with dimentions 5.30 mm by 5.30 mm.

The methodology outlined in this work incorporates an approach known as multitask learning. Multitask learning is a paradigm that seeks to enhance model training efficiency and scalability by leveraging commonalities across various tasks. This approach is particularly advantageous in scenarios where data is scarce, as it utilizes shared features among different tasks to facilitate more effective training and to achieve superior model performance by exploiting the inherent patterns present within the data [Caruana 1997, Baxter 1997, Ruder 2017]. Thus, with the multitask learning framework, the learning process is designed for the simultaneous detection of ventricular arrhythmias (VAs) and also the estimation of RR intervals from one-channel intracardiac electrograms (IEGMs). This approach is predicated on the rationale that engaging the model in concurrent learning tasks—specifically, VA detection and RR interval regression—enhances its performance on each task by leveraging the intrinsic relationship between them. The selection of this approach is driven by the goal of deploying the model on an embedded system (NUCLEO-L432KC) with stringent memory constraints, necessitating the use of compact models in terms of parameter size. The additional task of RR interval prediction is posited to facilitate the model in learning crucial patterns relevant to cardiac rhythm, given that the arrhythmias in focus are intimately linked to heart rate variability.

The proposed approach, designed for the STM32L432, reached a detection capability with an F_β score of 0.88, an overall accuracy of 93.55%. The inference latency is small with 59.96 ms, and demonstrated efficient resource utilization with memory consumption well within the microcontroller's capacity.

This paper is structured as follows: Section 2 reviews the literature, establishing the groundwork for our approach. Section 3 articulates our proposed method. The experimental procedures and results discussion are outlined in Section 4. Section 5 offers conclusions and suggests potential research trajectories, charting a course for further inquiry in the field.

2. Related Works

Recent progress in arrhythmia detection technologies has been advanced through the application of multi-task learning and deep learning, particularly in the context of wearable and embedded systems. In this context, we explore four foundational studies that have been crucial for the development of this work.

In the first study, [Soto and Ashley 2020], the authors introduce a method that uses multi-task learning to evaluate the signal quality and detect arrhythmias in wearable devices simultaneously. The system's performance metrics, achieving precision and recall rates of 0.94 and 0.98 respectively for atrial fibrillation detection, highlight its effectiveness. The methodology leverages unsupervised transfer learning with convolutional denoising autoencoders, significantly outperforming traditional single-task approaches.

In [Medhi et al. 2023], the researchers extend multi-task learning applications across various ECG databases. The study demonstrates an accuracy enhancement of up to 5.2% on the MIT-BIH arrhythmia database, validating the benefits of multi-task learning in complex bioelectrical signal analysis scenarios. The methodology employs a detailed comparative analysis across different datasets, underscoring the flexibility and robustness of the proposed multi-task learning framework.

In [Torres-Soto and Ashley 2020], the authors further affirm multi-task learning's potential in cardiac health monitoring, specifically within wearable device contexts. Achieving high metrics—sensitivity of 0.98 and specificity of 0.99—this research delineates a robust framework for atrial fibrillation detection, leveraging the inherent data complexity in wearable devices for nuanced and accurate classification.

It is explored in [Geng et al. 2023] the integration of a Contextual Transformer (CoT) block within a multi-task learning schema, evaluated on the CPSC2018 and PTB-XL datasets. Achieving F1 scores of 0.827 and 0.833 respectively, this approach underscores the CoT block's ability to dynamically model ECG sequences for enhanced classification, offering a substantive advance in ECG analysis precision.

Incorporating RR interval analysis in a multi-task learning model provides essential temporal information about heart mechanics. This integration improves the model's accuracy, offers a deeper understanding of heart dynamics over time, and enhances its ability to generalize and handle variability in data, making it more effective for diagnosing and monitoring cardiac conditions. This approach allows the model to capture both the static and dynamic aspects of cardiac function, resulting in more accurate predictions, particularly in diagnosing and monitoring cardiac conditions.

3. Methodology

In this section, we explain the methodology adopted for multitasking in the context of IEGM recordings. Utilizing a custom-built deep learning architecture, we integrate multitask learning strategies to simultaneously address various aspects of cardiac signal interpretation. The methodology is twofold, data handling and multitask neural network process. The neural network architecture is designed to be compact with a low footprint, facilitating integration into the NUCLEO-L432KC where resources are limited.

3.1. IEGM Database

The IEGM database is constructed based on the IEGM recordings provided by SingularMedical [Suzhou Singular Medical Co., Ltd. 2023], which was used in the 2022 ACM/IEEE TinyML Design Contest [Jia et al. 2023]. Intracardiac electrogram (IEGM) recordings from 90 patients with single-chamber ICDs, captured at a sampling rate of 250 Hz, were segmented and analyzed. The recordings were segmented using a 5-second sliding window approach, with 150 points offset at each step, yielding segments comprising 1250 data points. Each 5-second signal segment provided a detailed representation of the heart's rhythm during that interval, enabling a thorough categorization of the cardiac conditions.

These segments were classified into two primary groups: ventricular arrhythmias (VAs) and non-VAs, which were further subdivided into eight specific rhythm categories: Atrial Fibrillation (AFb), Atrial Flutter (AFt), Sinus Rhythm (SR), Supraventricular Tachycardia (SVT), Ventricular Fibrillation (VFb), Ventricular Flutter (VFt), Ventricular Premature Depolarization (VPD) and Ventricular Tachycardia (VT). For the category of life-threatening VAs, the corresponding sub-category labels include VT, VFb and VFt. For the non-VAs, the segments are labeled with a sub-category label other than VT, VFb, and VFt. Figure 2 presents two signals. The one above is a Non-VA beat and the other is a VA one.

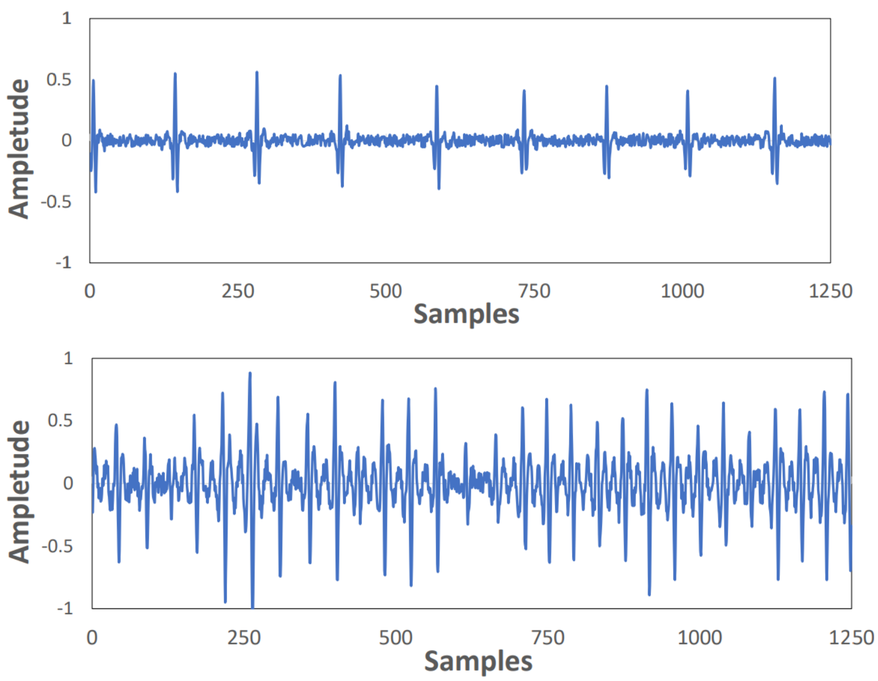

Figure 2. Representation of a non-VA signal (figure above) and a VA signal (figure below). Source: adapted from [Jia et al. 2023].

To validate the robustness of AI/ML algorithms in practical, real-world scenarios, these IEGM signals were allocated into training and testing sets on a patient-wise basis.

This approach ensured no patient data was present in both training and test sets, with 85% of the signals designated for algorithm training and the remaining 15% set aside for evaluation, according to the training and testing data already provided by the 2022 ACM/IEEE TinyML Design Contest [Jia et al. 2023].

3.2. Data Handling and Preprocessing

The IEGM database is comprised of single-lead recordings from RVA-Bi leads of ICD devices, each being extended over 5 seconds and sampled at a rate of 250 Hz, resulting in 1,250 data points per recording being obtained. While labels for various classes of arrhythmic beats are provided within this dataset, specific information on RR intervals is not included. Given the necessity of RR interval data for the secondary task in the multitask learning approach, this information is required to be computed for the training dataset. The well-known Pan-Tompkins algorithm [Pan and Tompkins 1985] is utilized for peak detection in this context. The algorithm is divided into several stages: (i) bandpass filtering is applied to isolate the frequencies characteristic of QRS complexes, (ii) differentiation is performed to enhance the slopes of the QRS segments, (iii) squaring is done to intensify peak signals, (iv) moving-window integration is employed to consolidate these peaks, and (v) adaptive threshold, coupled with decision logic, is used for precise QRS peak identification. Additional refinement steps are taken to rectify potential errors, thereby ensuring the reliable detection of QRS complexes under diverse conditions.

3.3. Multitasking

In the domain of machine learning, particularly in the context of neural network designs, Multitask Learning (MTL) [Ruder 2017] emerges as a strategy that enables a model to address multiple learning tasks simultaneously. Unlike traditional single-task learning approaches that develop separate models for each task, multitask learning advocates for a unified approach, optimizing a single model across various tasks. This learning strategy tends to improve learning efficiency, elevate model generalization capability, and enhance task-specific accuracy [Ruder 2017].

In the context of IEGM signal analysis, the multitask approach is designed for a convolutional neural network. An initial series of shared convolutional layers, which extract and refine signal features, is followed by the network branching out, with distinct pathways being dedicated to different but related tasks: one for the classification of arrhythmias and another for the quantification of a regression task: inference of RR-interval duration.

The general scheme of multitask learning is illustrated in Figure 3, showcasing how shared and task-specific learning components interact within the network to foster a richer understanding of the data. In this work, only the final class for classification is specific, all the other layers are shared over all tasks.

3.4. IEGMNet

The work presented in [Hannun et al. 2019] demonstrated the efficacy of compact neural network architectures in arrhythmia detection, which aligns with our goal to implement a model with a limited number of convolutional layers. By adopting a multitask learning strategy within a lightweight neural network, we aim to improve the model's predictive accuracy and ability to learn ECG signal patterns.

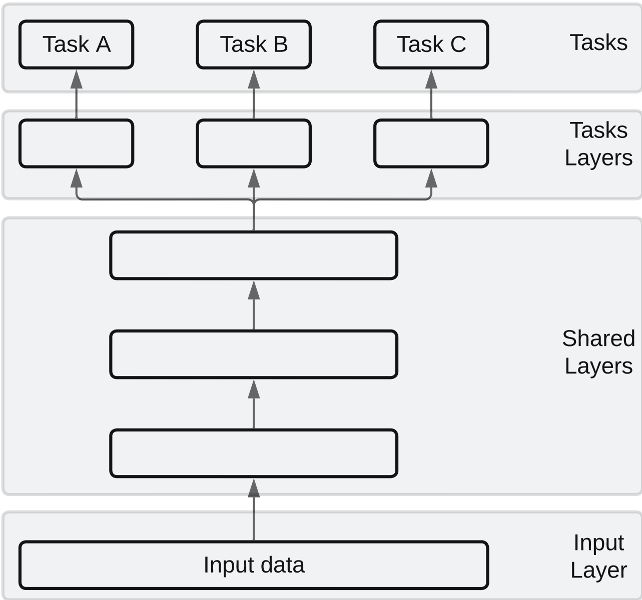

Figure 3. A general scheme of multitask learning. Source: adapted from [Ruder 2017].

In this context, the IEGMNet is a convolutional neural network (CNN) designed to process IEGM segments from ICD devices. The structure and Layers of IEGMNet is presented in Figure 4.

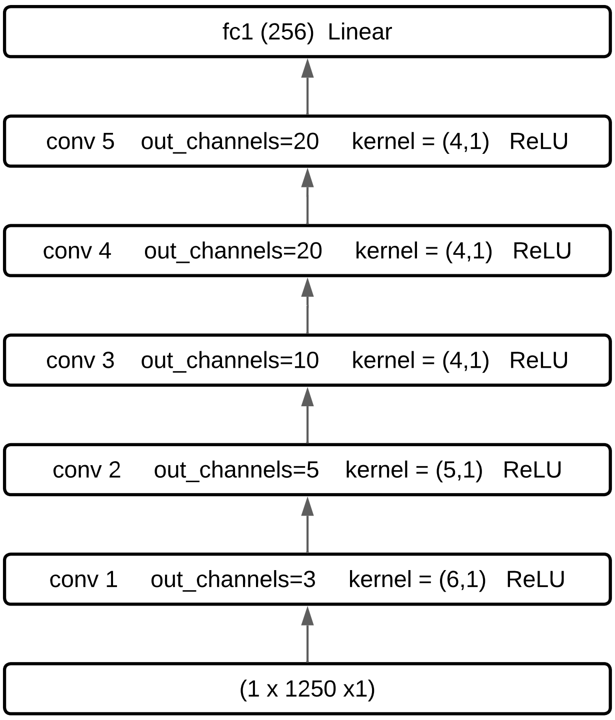

Figure 4. Architecture of the IEGMNet.

The IEGMNet can be summarized as follows:

- **Network Input:** IEGMNet accepts as input a one-dimensional tensor representing a fixed-size IEGM segment (1250 data points). This data is reshaped into an input tensor with dimensions [1, 1250, 1], preparing it for convolutional processing.
- **First Convolutional Layer (conv1):** This layer employs (6, 1) convolutional filters to extract features from the input signal. It increases the input tensor's depth to three channels. ReLU activation and batch normalization are applied to enhance training stability and efficiency.
- **Second Convolutional Layer (conv2):** Uses (5, 1) filters to deepen to five channels. Continues with ReLU and batch normalization.
- **Third Convolutional Layer (conv3):** Utilizes (4, 1) filters, increasing the depth to ten channels. Maintains ReLU and batch normalization.
- **Fourth Convolutional Layer (conv4):** Employs (4, 1) filters, expanding depth to twenty channels. Again applies ReLU and batch normalization.
- **Fifth Convolutional Layer (conv5):** Keeps (4, 1) filters and depth at twenty channels. Uses ReLU and batch normalization.
- **First Fully Connected Layer (fc1)**: After convolutional layers, the network flattens the tensor and processes data through a dense (fully connected) layer. Incorporates dropout for regularization and overfitting reduction. Outputs a 256-element feature vector.

3.5. Hydranet: A multitasking network

IEGMNet is selected as the foundational architecture for our analytical methodology, specifically designed for the analysis of IEGM signals following preprocessing. This convolutional neural network (CNN) is characterized by a sequence of convolutional layers, which are followed by the application of ReLU activation and Batch Normalization.

Based on IEGMNet, the HydraNet is further developed as an architecture that adopts multi-task learning to cope with this work analysis. HydraNet is an evolution of IEGMNet, characterized by its division into two distinct branches after the feature extraction phase, each targeting a specific output: one branch is dedicated to Arrhythmia Classification (AC), focusing on discerning arrhythmic patterns, and the other to Regression-Related metric (RR-interval prediction). This bifurcation allows the model to leverage a wider spectrum of signal insights, thereby facilitating a more comprehensive interpretation of IEGM data. Figure 5 displays the proposed multitask HydraNet architecture.

The HydraNet takes the IEGMNet 256-feature output vector as input to each branch. Each HydraNet's branch is a sequence of dense (fully-connected) layers and a specific activation according to the task-specific output. The RR branch uses a linear layer followed by ReLU activation and another linear layer to produce the final output, employing a Mean Absolute Error (MAE) loss function suitable for regression tasks. Similarly, the AC branch uses a linear layer followed by ReLU and another linear layer, utilizing binary cross-entropy loss to the classification task.

The training process for HydraNet integrates a unified loss function addressing both classification and regression objectives. To formally define the loss function of HydraNet, consider the network's output for two tasks: Arrhythmia Classification (AC)

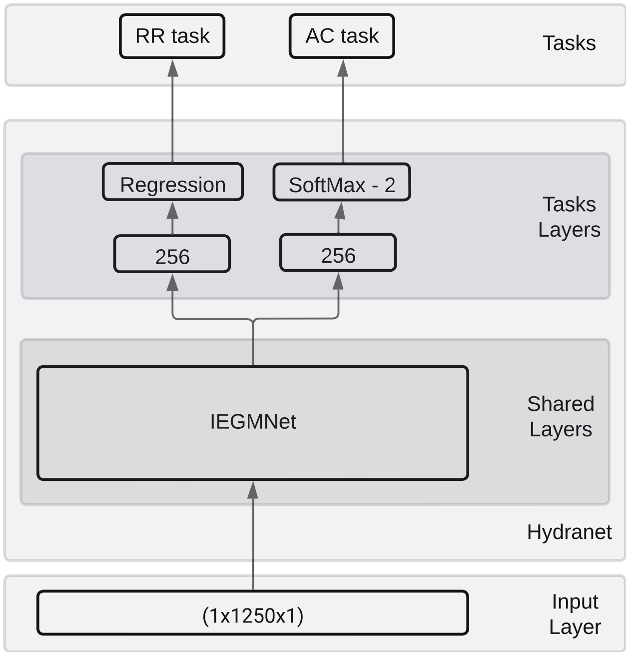

Figure 5. Proposed multitask Hydranet.

and Regression Task (RT) metric estimation, such as the RR interval. Let y_{AC} denote the true label for the arrhythmia classification task, and y_{RT} represent the true value for the regression task (e.g., the RR interval). Correspondingly, let $\hat{y}_{AC}$ and $\hat{y}_{RT}$ denote the predicted outputs for these tasks.

The loss function for HydraNet can be defined as a weighted sum of the loss functions for the AC and RT tasks. For the AC task, the binary cross-entropy loss is employed, denoted as L_{AC}, which is suitable for binary classification tasks. For the RT task, the L1 loss function is selected, denoted as L_{RT}, which is appropriate for regression tasks due to its robustness to outliers. The overall loss function, L, for HydraNet can thus be defined as:

$$L = \lambda_{AC} L_{AC}(y_{AC}, \hat{y}_{AC}) + \lambda_{RT} L_{RT}(y_{RT}, \hat{y}_{RT}), \tag{1}$$

wherein λ_{AC} and λ_{RR} are weighting coefficients that adjust the relative importance of each task's loss to the overall cost function. L_{AC} can be defined as:

$$L_{AC}(y_{AC}, \hat{y}_{AC}) = -\left[y_{AC} \log(\hat{y}_{AC}) + (1 - y_{AC}) \log(1 - \hat{y}_{AC})\right] \tag{2}$$

which represents the binary cross-entropy loss for the AC task, and L_{RT} defined as:

$$L_{RT}(y_{RT}, \hat{y}_{RT}) = |y_{RT} - \hat{y}_{RT}| \tag{3}$$

which represents the MAE loss for the RR regression task.

This formulation allows HydraNet to optimize for both the AC and RT tasks simultaneously, leveraging shared features and insights to improve performance on both tasks. The choice of λ_{AC} and λ_{RT} can be determined empirically to balance the contributions of each task to the overall learning objective.

3.6. Hardware Platform and Design Setup

The experiments were conducted on a NUCLEO-L432KC development board equipped with an ARM Cortex-M4 processor operating at 80 MHz, backed by 256 KiB of flash storage and 64 KiB of SRAM. Additionally, it includes an integrated ST-LINK/V2-1 for efficient debugging and programming. Notably, the board's operational power consumption is approximately 30 mW, dropping to 1.5 mW in idle mode, making it an excellent choice for energy-sensitive embedded applications, including those in ICDs.

Further enhancing its utility, the board is compatible with the STM32 X-Cube-AI, an extension of the STM32Cube that facilitates the deployment of AI applications. This feature enables the automatic translation of pre-trained AI algorithms, encompassing both neural networks and traditional machine learning models, for direct integration into the project. The efficiency and lightweight design of the IEGMNet architecture are especially beneficial for the constrained environments of micro-controllers, ensuring that sophisticated AI models can be deployed without exceeding the limited memory and processing resources.

3.7. Evaluating metrics

The proposed algorithm is evaluated using three parameters: F_β, Inference Latency, and Memory Occupation. Equation 4 presents the F_β score, which computes the confusion matrix of the classification generated by the algorithm on the testing database. The case positive is VAs:

$$F_\beta = \left(1+\beta^2\right) \times \frac{\text{Precision} \times \text{Recall}}{(\beta^2 \times \text{Precision}) + \text{Recall}} \tag{4}$$

with $\beta = 2$.

This configuration prioritizes recall to enhance the detection of critical VAs, which is essential for ICD functionality. By emphasizing recall, the goal is to capture as many VA events as possible, minimizing the risk of missing any life-threatening conditions. Simultaneously, the strategy strives to maintain precise identification of non-VA instances, thus minimizing the potential for administering inappropriate shocks, thereby optimizing patient safety and the dependability of the ICD system.

For assessing practical performance, we evaluate Inference Latency and Memory Occupation. Specifically, Inference Latency, denoted as L in milliseconds, quantifies the response time of the algorithm when processing data on the NUCLEO-L432KC, reflecting its efficiency during real-world operation. This latency metric L is subsequently normalized according to Equation 5 to provide a standardized measure.

$$L_n = \left(1 - \frac{L - \text{Min}_L}{\text{Max}_L - \text{Min}_L}\right) \tag{5}$$

where $\text{Min}_L = 1ms$ and $\text{Max}_L = 200ms$.

4. Experiments and Results

In this section, we detail the experiments designed to test our approach concerning the feasibility of deploying a multitask architecture model within a common ICD embedded system, that has limited processing capabilities and memory.

The evaluation of the inference performance of the trained model is conducted on the NUCLEO-L432KC. However, the training is conducted on a machine with Intel i9-10900, along with 128GB DDR4 RAM and an RTX 3090 GPU with a dedicated memory of 24GB GDDR6X. For model manipulation, we used the PyTorch framework version 1.10.1 and Scikit-Learn version 1.2.1 for metric calculation.

The model was trained for 20 epochs, initial learning rate of 0.0001 and the Adam optimizer.

4.1. Results and Discussion

The computed F_β score with $\beta = 2$ stands at 0.88, reflecting the classification capability, particularly by emphasizing recall in the context of ventricular arrhythmia detection. The importance of this score lies in its consideration of both the precision and the sensitivity of the algorithm, with a greater emphasis on minimizing the possibility of missed life-threatening arrhythmias.

The confusion matrix presented in Figure 6 illustrates the model performance on the test set. With 3048 true negatives and 2178 true positives, the model shows the ability to correctly classify non-VA and VA events respectively. However, it is important to note the presence of 188 false positives and 211 false negatives, which indicate instances where the model has incorrectly classified the events. Despite these discrepancies, the model achieves an accuracy of 93.55%, which indicates its overall reliability.

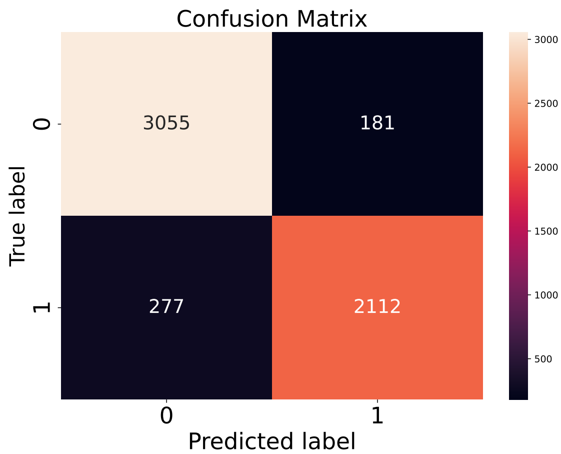

Figure 6. Model's confusion matrix

The latency of 59.96 ms in model inference and the total flash memory consumption of the algorithm stands at 225.98 KiB is significant concerning since its close to the STM32's memory capacity of 256 KiB. The efficient memory usage highlights the algorithm's compatibility with the STM32 microcontroller, which is known for its use in medical devices including defibrillators due to its reliability and computational power. The memory footprint is especially critical since it allows for additional functionalities and updates to be accommodated within the same hardware, future-proofing the device's capability.

In summary, the detailed results underline the algorithm's robustness and efficiency in detection, speed, and memory usage, affirming its suitability for integration into STM32-based ICDs where accurate and timely detection of VAs is vital for patient safety.

5. Conclusion

This study explored the implementation of neural networks in implantable cardioverter defibrillators (ICDs) for effective arrhythmia detection within resource-constrained environments. The aim of this work is to verify whether an efficient, multitask learning-based neural network could operate within the typical processing environment of an ICD, specifically the NUCLEO-L432KC. The experimental results demonstrated that the proposed multitask learning model, which combines ventricular arrhythmia detection with RR interval regression, was able to diagnose arrhythmias within a compact, resource-efficient framework, achieving an F_β score of 0.88 and an accuracy of 93.55%, with a inference latency of 59.96 ms. These results demonstrate the feasibility of integrating the proposed multitask neural network within ICDs.

For future work, we aim to explore the impact of quantization on larger architectures. The process of quantization, which reduces the precision of the network parameters, offers the potential for significant reductions in model size, making it particularly appealing for deployment on resource-constrained devices. By investigating how larger neural network architectures respond to quantization, we seek to identify optimal trade-offs between model complexity, performance, and efficiency.

Acknowlegment

The authors would also like to thank the *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior* - Brazil (CAPES) - Finance Code 001, *Fundacão de Amparo à Pesquisa do Estado de Minas Gerais* (FAPEMIG, grants APQ-01518-21), *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq, grants 308400/2022-4) and Universidade Federal de Ouro Preto (PROPPI/UFOP) for supporting the development of this study.

References

Baxter, J. (1997). A bayesian/information theoretic model of learning to learn via multiple task sampling. Machine Learning, 28(1):7–39.

Caruana, R. (1997). Multitask learning. In Machine Learning Proceedings 1997, pages 41–48. Elsevier.

DiMarco, J. P. (2003). Implantable cardioverter–defibrillators. New England Journal of Medicine, 349(19):1836–1847.

Geng, Q., Liu, H., Gao, T., Liu, R., Chen, C., Zhu, Q., and Shu, M. (2023). An ecg classification method based on multi-task learning and cot attention mechanism. In Healthcare, volume 11, page 1000. MDPI.

Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., and Ng, A. Y. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature medicine, 25(1):65–69. Publisher: Nature Publishing Group.

Jia, Z., Li, D., Liu, C., Liao, L., Xu, X., Ping, L., and Shi, Y. (2023). Tinyml design contest for life-threatening ventricular arrhythmia detection. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.

Luz, E. J. d. S., Schwartz, W. R., Cámara-Chávez, G., and Menotti, D. (2016). ECG-based heartbeat classification for arrhythmia detection: A survey. Computer methods and programs in biomedicine, 127:144–164. Publisher: Elsevier.

Madhavan, M. and Friedman, P. A. (2013). Optimal programming of implantable cardiac-defibrillators. Circulation, 128(6):659–672.

Medhi, J. K., Ren, P., Hu, M., and Chen, X. (2023). A deep multi-task learning approach for bioelectrical signal analysis. Mathematics, 11(22):4566.

Mirowski, M. (1985). The automatic implantable cardioverter-defibrillator: an overview. Journal of the American College of Cardiology, 6(2):461–466.

Mousavi, S. and Afghah, F. (2019). Inter-and intra-patient ecg heartbeat classification for arrhythmia detection: a sequence to sequence deep learning approach. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 1308–1312. IEEE.

Myerburg, R. J., Kessler, K. M., and Castellanos, A. (1992). Ventricular fibrillation and sudden cardiac death. New England Journal of Medicine, 326(11):741–747.

Pan, J. and Tompkins, W. J. (1985). A real-time qrs detection algorithm. IEEE transactions on biomedical engineering, pages 230–236.

Physiopedia (2022). Ventricular fibrillation. `https://www.physio-pedia.com/Ventricular_Fibrillation`. Accessed: March 19 of 2024.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098.

Soto, J. T. and Ashley, E. (2020). Deepbeat: A multi-task deep learning approach to assess signal quality and arrhythmia detection in wearable devices. arXiv preprint arXiv:2001.00155.

STMicroelectronics (2020). Ultra-low-power with fpu arm cortex-m4 mcu 80 mhz with 256 kbytes of flash memory, usb. [Online]. Available: `https://www.st.com/en/microcontrollers-microprocessors/stm32l432kc.html`.

Suzhou Singular Medical Co., Ltd. (2023). Singular medical. `http://www.singularmedical.net/`. [Online]. Available.

Torres-Soto, J. and Ashley, E. A. (2020). Multi-task deep learning for cardiac rhythm detection in wearable devices. NPJ digital medicine, 3(1):116.

Zanker, N., Schuster, D., Gilkerson, J., and Stein, K. (2016). Tachycardia detection in icds by boston scientific: algorithms, pearls, and pitfalls. Herzschrittmachertherapie & Elektrophysiologie, 27(3):186.

Zipes, D. P. and Wellens, H. J. J. (1998). Sudden cardiac death. Circulation, 98(21):2334–2351.