Cardiac Anomaly Detection models for wearable devices

Aaptha B V* , Ankith B V*, Aryan Sharma*, Dr Madhura Purnaprajna**
*Department of Computer Science and Engineering
**Centre for Heterogeneous and Intelligent Processing Systems Research Centre
*PES University, Bangalore, India

***Abstract*—Sudden Cardiac Arrest (SCA) is a devastating heart abnormality which leads to millions of casualties per year. Thus, early detection or prediction of SCA could save human lives in a greater scale. In this work, we aimed to predict SCA before its occurrence and significant results has been obtained using the proposed signal processing methodology. Models were trained using a CNN, CNN + Long Short Term Memory (LSTM) model and a Random Forest Classifier on the MIT-BIH Arrhythmia dataset. Taking these models to an embedded device is the main purpose of this work. The CNN models were compressed using quantisation. Our early results indicated 96% on accuracy with approximately 33.16% reduction in size. We propose to apply variable width quantisation and retraining to further improve compression.**

***Index Terms*—Cardiac Anomaly, Deep learning, Healthcare, Machine learning, 2-lead ECG readings, waveform, time series data, Full integer quantisation.**

I. Introduction

In recent years, high performance computing has played an increasingly important role in scientific and engineering applications. One such application is the detection of cardiac anomalies. There are a few cardiac anomalies which, if detected early, can be treated successfully to save a person's life. But there are very few healthcare devices that prove to be efficient and convenient to use at the same time. Our solution is inclined towards helping people who are prone to getting cardiac arrests. By using ECG data, various machine learning and deep learning algorithms can be used to detect anomalous patterns. Many models regarding the same topic are not size-aware even though there are several available. Due to data sparseness, full-precision arithmetic is used and this leads to a large model size. In this paper, we propose different lightweight models that prove to be computationally less intensive and energy efficient.

II. Methodology

We have used PhysioNet's MIT-BIH Arrhythmia dataset [10] for our experiment. The MIT-BIH Arrhythmia Database contains 48 half-hour excerpts of two-channel ambulatory ECG recordings. An Arrhythmia dataset is enough to check for irregularities in a person's heart beat. Since we are not classifying the beats with respect to it's annotations, we did not check which abnormality occurs. The MIT-BIH dataset contains Electrocardiogram waveform data. In order to pre-process the data, we require specialized libraries to extract the information required. We use the wfdb package [12] for this purpose. First, we take a look into the Pyshiobank Beat annotations and classify them as normal and abnormal beats.

We then extracted the label information using the specified package and split the hour long excerpts into smaller chunks. The splitting of ECG data is done in order to get more training data. The dataset contains approximately 82,873 inputs after pre-processing. To detect these abnormalities, we came up with 3 solutions - Random Forest Classifier, CNN and a CNN+LSTM model.

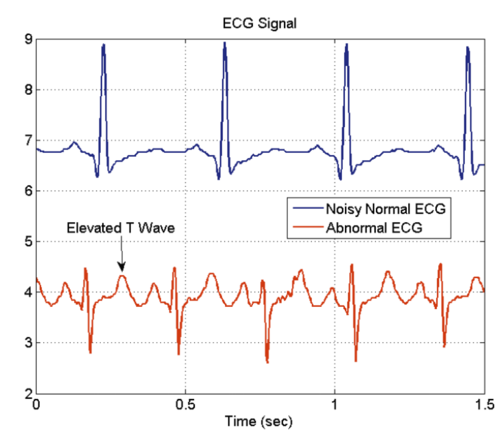

Fig. 1. Normal ECG Vs Abnormal ECG reading [13].

III. Experimental Setup

A. Convolutional Neural Network

We made a lightweight two layered convolutional neural network. The model inputs the ECG data and outputs the beat type (normal or abnormal). The network consists of a convolutional layer and a Dense layer including a flatten layer and dropout of 0.5. The accuracy of this model was approximately 96.19% with a validation accuracy of 81.43%.
FLOPS is a great way to find out the computational power of your model. Using Tensorflow libraries we calculated the Floating Operations Per Second (FLOPS) of our model and it came out to be 17,315.

B. Random Forest Classifier

Deep learning approaches are not always necessary. While they are more accurate on time series date, they are quite large in size compared to small machine learning algorithms. One such example is a Random Forest Classifier. Random Forest is an ensemble machine learning algorithm which is mainly used on structured data such as tabular data. But they can also be used to classify time series data. To apply RF algorithm on time series data, we transform it into a supervised learning problem. To do so, we do the pre-processing mentioned in previous section, but instead of feeding these inputs into a neural network, we input these values into a csv file. This file was read and the random forest classifier was implemented on it.

C. CNN + LSTM model

LSTMs can handle sequential data more efficiently than CNNs, which are small and efficient, but not as efficient during processing. Our results could be confirmed by using a combination of convolutional neural network and a recurrent neural network architecture called Long Short Term Memory (LSTM) to determine whether they corresponded to our data more closely. We used one convolutional layer, a dense layer and a LSTM layer of 2 inputs in a TimeDistributed architecture to obtain our model. The accuracy of LSTM model was approximately 98.49% with a validation accuracy of 80.81%. We had around 51 thousand parameters with 86,621 FLOPS.
Originally our model was 202.86 KB. But after changing the filter and kernel size, our model size dropped down to 67.67 KB. We did full integer quantisation on this model and the size reduced by 4 times, speedup increased by 3 times. This quantisation is the most essential method in order to make a model work on a microcontroller. Our model size became 16.92 KB. Post-training quantisation is a conversion technique that can reduce model size while also improving CPU and hardware accelerator latency, with little degradation in model accuracy. We obtained an accuracy of 97.96%. The difference between the accuracy's of both the models was merely 0.53%.

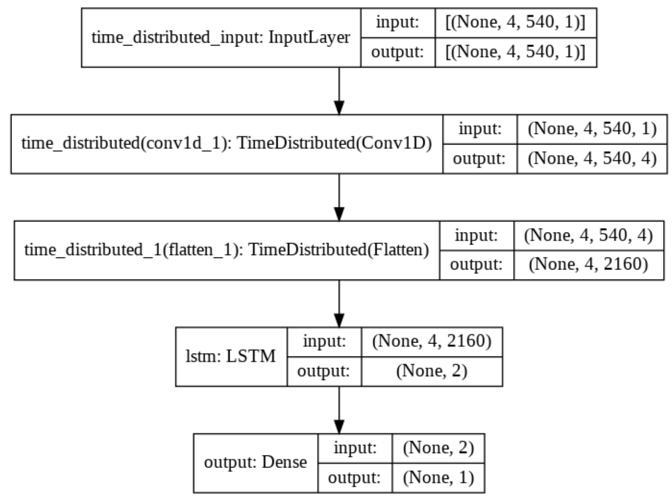

Fig. 2. CNN + LSTM model architecture.

IV. Experiment Results

As shown in Table I, we quantised the performance of all the models on the MIT-BIH Arrhythmia dataset and tabulated its accuracy. We checked whether the models were overfitting by comparing the performance on training, testing, and validation datasets. The accuracy on training data was high when compared to the test/val accuracy. To avoid this we increased the dropout rate by 0.2. The accuracy on the validation set increased by 9.26%. However, the random forest classifier did not overfit the dataset.

TABLE I
Accuracy scores of methods on the MIT-BIT Arrhythmia Dataset

Architecture	**Acc. [%]**	**Val. Acc. [%]**	**Param #**	**FLOPS**
CNN	96.19	81.43	17,321	17,315
CNN + LSTM	98.49	80.81	51,932	86,621
CNN (Q)	96.13	86.42	17,321	17,315
CNN + LSTM (Q)	97.96	83.16	51,888	69,299
Random Forest	85.8	85.8	-	-

Note: Q is a quantised model

Full integer quantisation on the models increased the speedup by 3 times. We can further improve the usability of these models in real life situations by combining the

Arrhythmia dataset with other potential datasets such as an atrial fibrillation dataset. This is because atrial fibrillation can cause serious harm and may result in brain strokes.

Conclusion and Related Work

This paper presents a novel and lightweight neural network architecture that can be implemented on embedded devices. While other implementations of the same exist [1] [2], our primary objectives were to minimize the memory footprint and propose a computationally less intensive approach to detect cardiac anomalies. Using Lightweight CNNs and a Random Forest classifier presented the desired results. We observe that Random Forest classifiers can be considered in the detection of cardiac anomalies. This opens up the possibility of using Machine Learning models over Deep Learning models.

References

[1] Pranav Rajpurkar, Awni Y. Hannun, Masoumeh Haghpanahi, Codie Bourn and Andrew Y. Ng. ”Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks”. CoRR abs/1707.01836, 2017b.

[2] Thorir Mar Ingolfsson, Xiaying Wang , Michael Hersche, Alessio Burrello, Lukas Cavigelli and Luca Benini. ”ECG-TCN: Wearable Cardiac Arrhythmia Detection with a Temporal Convolutional Network”. CoRR abs/2103.13740, 2021b.

[3] Saadatnejad, Saeed and Oveisi, Mohammadhosein and Hashemi, Matin. ”LSTM-Based ECG Classification for Continuous Monitoring on Personal Wearable Devices”. vol. 24, no. 2, pp.515-523, 2020.

[4] Tithi, Sushmita Roy and Aktar, Afifa and Aleem, Fahimul and Chakrabarty, Amitabha. ”ECG data analysis and heart disease prediction using machine learning algorithms”. pp.819-824, 2019.

[5] Bonizzi, Pietro and Driessens, Kurt and Karel, Joel. ”Detection of atrial fibrillation episodes from short single lead recordings by means of ensemble learning”. pp.1-4, 2017.

[6] Karimifard, S. and Ahmadian, A. and Khoshnevisan, M. and Nambakhsh, M. S. ”Morphological Heart Arrhythmia Detection Using Hermitian Basis Functions and kNN Classifier”. pp.1367-1370, 2006.

[7] Robert David, Jared Duke, Advait Jain, Vijay Janapa Reddi, Nat Jeffries and, Jian Li, Nick Kreeger, Ian Nappier, Meghna Natraj, Shlomi Regev, Rocky Rhodes, Tiezhen Wang and Pete Warden. ”TensorFlow Lite Micro: Embedded Machine Learning on TinyML Systems”. CoRR abs/2010.08678, 2020b.

[8] Ariel Gordon, Elad Eban, Ofir Nachum, Bo Chen, Tien-Ju Yang and Edward Choi. ”MorphNet: Fast & Simple Resource-Constrained Structure Learning of Deep Networks”. CoRR abs/1711.06798, 2017b.

[9] Majumder, AKM Jahangir, ElSaadany, Yosuf, Young, Roger and Ucci, Donald. ”An Energy Efficient Wearable Smart IoT System to Predict Cardiac Arrest”. pp. 1-21. 2019b.

[10] Moody, G.B. and Mark, R.G. ”The impact of the MIT-BIH Arrhythmia Database”. Vol. 20, no. 3, pp.45-50. 2001b.

[11] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng CK, Stanley HE. ”PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals”. vol. 101, No. 23, pp. e215-e220, 2000.

[12] Xie, Chen, McCullum, Lucas, Johnson, Alistair, Pollard, Tom, Gow, Brian, and Benjamin Moody. ”Waveform Database Software Package (WFDB) for Python (version 3.3.0). PhysioNet”. Year 2021.

[13] Image retrieved from https://lh4.googleusercontent.com.