Arrhythmia classification using a cascade of binary classifiers based on multi-branch convolutional Neural Network

Nicola Snoriguzzi[†], Alberto Zancanaro[‡]

***Abstract*—An electrocardiogram (ECG) measures the electric activity of the heart and it has been widely used to detect heart diseases due to its simplicity and non-invasive nature.**

The main difficulties in the automatic recognition of arrhythmias are due to the great variability of the signal which often requires manual intervention.

In this study, we propose a binary classification approach that uses multi-branch Neural Network, capable of combining information extracted from datasets containing different representations of the signal.

Data come from the PhysioNet/Computing in Cardiology Challenge 2017 and they consist in single lead ECG recordings, collected using the AliveCor device, lasting from 9 s to just over 60 s.

***Index Terms*—ECG, Arrhythmia classification, CNN, binary classifier, multi-branch network, peak extraction**

I. Introduction

Electrocardiogram (ECG) recording is an important clinical tool for detecting cardiac disorders. It consist in recording the electrical activity of the heart in time through various electrode placed on the skin. An ECG recording could lasts from a few seconds to multiple days and its precision depend to the number of electrode use (e.g. from simple portable device that use only one or to channel to register to more professional device that use various channel).

To identify the heart beats it is necessary to recognize the various parts that characterized each beat, so that it can be subsequently isolate and analyse. As shown in Fig. 1, it is possible to recognize several significant points within a heartbeat, with which many different characteristics can be extracted. They are summarized in Fig. 1.

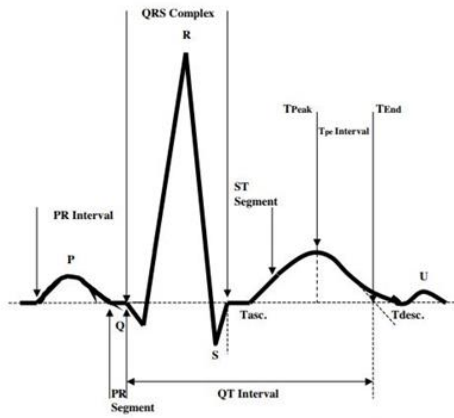
An arrhythmia is an irregularity in the rate or rhythm of the heartbeat. During an arrhythmia, the heart can beat too fast, too slow, or with an irregular rhythm [2].

Between the cardiac pathologies that can be reveled from ECG, atrial fibrillation (AF) is the most prevalent and can occur in sustained or intermittent episodes. These two states make the diagnosis of AF challenging, particularly when only short ECG (in the order of seconds) are available.

Despite ECG being a well-established method, the classification of arrhythmic or others type of problems is generally performed in a manual or semi-automated manner by cardiologists, who review each signal in the search for abnormalities.

[†]email: nicola.snoriguzzi@studenti.unipd.it

[‡]email: alberto.zancanaro.1@studenti.unipd.it



Feature	Description	Duration
RR	interval between R wave and the next R wave	0.6-1.2 s
P	first short upward movement of the ECG	80ms
PR	measured from the beginning of the P wave to the beginning of the QRS complex	120-200 ms
QRS	normally begins with a downward deflection Q, a larger upwards deflection R and ends with a downward S wave	80-120 ms
PR	connects the P wave and the QRS complex	50-120 ms
J-point	The point at which the QRS complex finishes and the ST segment begins is called J-point.	Not applicable
ST	connects the QRS complex and the T wave	80-120 ms
T	normally a modest upward waveform	160 ms
ST	measured from the J point to the end of the T wave	320 ms
QT	measured from the beginning of the QRS complex to the end of the T wave	420 ms
U	normally has low amplitude and often it is completely absent	Not mentioned

Fig. 1: Normal ECG waveform and table with features and their normal duration. (Image taken from [1]).

The process is therefore slow, prone to mistakes caused by human factors, expensive and suffers from inter- and intra-rater variability [3].

Classical approaches to ECG signals automatic classification are based on the use of various hand-engineered features. Examples of this feature could be heart rate variability (HRV) metrics [4] and morphological characteristics (e.g. P-wave absence) [5].

The advancements of the last year in the field of the machine learning (e.g. convolutional neural networks, deep learning and so on) can provide us net tool to face the difficulties of automated ECG classification. Deep learning methods were especially useful in this task due to their ability to *automatically learn features* at multiple levels of abstraction (i.e. layers). This allows the system to learn complex functions by mapping the input to the output directly from data without depending on hand-engineered features [6].

Methods based on hand-engineered features are generally more easy to implement at computational level but required and extensive work of pre processing and feature extraction. On other hands methods based on deep learning require very little preprocessing but are computationally expensive and the features used lack any direct link with the real characteristic of the signal.

In this study we will focus on a new procedure that tries to balance the strengths and weaknesses of both approaches. We benchmark a cascade of binary classifiers based on a multi-branch network fed with two different datasets extracted using time and frequency analysis of short ECG segments provided for the Physionet/Computing in Cardiology Challenge 2017 [7].

II. Related Work

The PhysioNet 2017 offers a wide pool of publications about automatic classification of ECG signals.

Various works are focused on the direct classification using a Neural Network, e.g. Zihlmann et al. [8]. Other approach, like the one explained in the Chengyu Liu et al. [9], is based on a simpler classifier (Support-Vector Machine SVM) combined with a more complex features extraction like the R-peak detection of Pan and Tompkins [10].

There are also various works that try to combine the use of Neural Network with a complex feature extraction. In this category there are two main approaches. The first uses Neural Network to extract features to use in simple classifier. The second one uses hand-engineered features (QRS period etc) as input of a Neural Network [11].

III. Processing Pipeline

The early stage of the work was focused in the development of a classifier that use a single CNN to categorize the ECG but this approach led to two main problems: the network had a very *low level of accuracy* (around 40%) and it was very complex and time consuming to train.

The low accuracy was due to a lack of the network to extrapolate enough features from a single dataset. This led to the creation of different datasets and multi-branch network.

Another strategy used to improve the accuracy was to use the Network for a simpler binary recognition task, i.e. recognize if an ECG belongs or not only to a specific category, and subsequently assign the final tag combining the results of the binary classifiers.

Brief description of our work:

- dataset creation, feature extraction using peak detection, segmentation and time/frequency analysis
- use of multi branch CNN network
- binary approach

IV. Signals and Features

The data used for this project are taken from the PhysioNet/-Computing in Cardiology Challenge 2017 [7], in particular we have focused on the first 6000 records of the dataset. The dataset contains 8,528 single lead ECG recordings, collected using the AliveCor device, lasting from 9 s to just over 60 s. ECG recordings were sampled as 300 Hz and they have been band pass filtered by the AliveCor device. All data are provided in MATLAB V4 WFDB-compliant format (each including a .mat file containing the ECG and a .hea file containing the waveform information). Figure 2 shows the examples of the ECG waveforms (lasting for $20s$) for the four classes in this Challenge. From top to bottom, they are ECG waveforms of normal rhythm, AF rhythm, other rhythm and noisy recordings. Figure 3 summaries some information of the PhysioNet dataset.

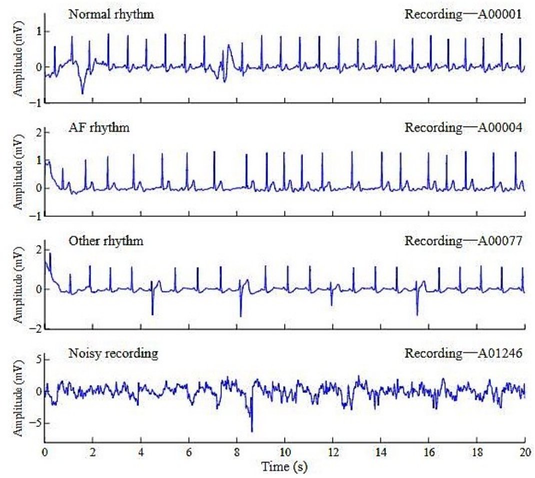


Fig. 2: Examples of ECG waveforms. (Figure taken from [7])

Type	# recording	Time length (s)				
		Mean	SD	Max	Median	Min
Normal	5154	31.9	10.0	61.0	30	9.0
AF	771	31.6	12.5	60	30	10.0
Other rhythm	2557	34.1	11.8	60.9	30	9.1
Noisy	46	27.1	9.0	60	30	10.2
Total	**8528**	**32.5**	**10.9**	**61.0**	**30**	**9.0**

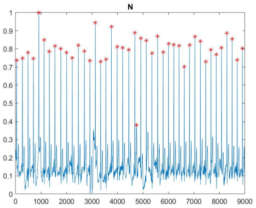
Fig. 3: Data profile for the dataset. (Figure taken from [7])

Feature extraction is one of the most important and delicate phases in a learning algorithm based on Neural Network (NN), because the structure of the dataset ad the information represented inside could largely influence the accuracy and the training time of the NN.

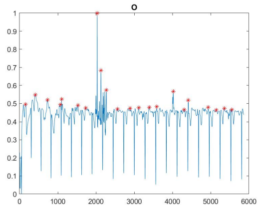
We have used MATLAB to analyze and process the signal in order to create different datasets used in a second stage to classify the trials into the corresponding arrhythmia category. In particular, we have identified two datasets containing different signal features in order to increase the descriptive capacity of the data that will be provided to classifiers.

A. Preprocessing and data augmentation

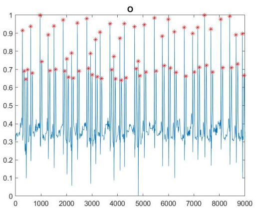
As you can see from table 3, the dataset is not balanced because there are very few noisy trials compared with the occurrence number of the other categories. For this reason we have selected the noisy trials and we have created different versions of them; in particular we have quadruplicated the number of noisy trails using Gaussian noise with three different SNR values using the function *awgn(in,snr)*.



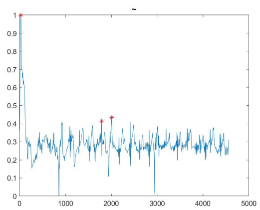
(a) trial A0193



(b) trial A0195



(c) trial A0198



(d) trial A0201

Fig. 4: Result of peak detection.

Finally, we have normalized all the trials in range $[0, 1]\ mV$.

In all, we have created six different representations of the signals, each saved in a separate dataset. Then we have performed several tests using different Neural Network structures (CNN, FCNN and RNN), in order to understand which dataset was the most significant. In conclusion, we have selected the two datasets described below because they obtained the better results.

B. *Dataset A*

First of all, we have found the R peaks of the ECG signal using a peak detector [12] (you can find some examples in Figure 4). Then we have considered a window of $500\ ms$ centered in the detected peaks, we have segmented the signals and we have plot all the peaks of each trial in a single black and white image of shape [150, 200]. At the end, Dataset A consists of a three dimension array [number of trials, 150, 200] with values in range [0,1]. Figure 5 shows some examples of Dataset A.

C. *Dataset B*

For dataset B we have considered a signal window of $15\ s$ and we have taken the central part of each record. If the record was shorter than $15\ s$, we have padded the signal in a symmetric way adding zeros on the right a on left. Then we have computed the spectogram s using the MATLAB function *spectrogram* using: $window = 80$, $noverlap = 50$ and $nfft = 256$. Finally, to create the dataset we have considered the normalization of $log(abs(s))$. At the end, Dataset B consists of a three dimension array [number of trials, 129, 140] with values in range [0, 1]. See Figure 6 for some examples.

V. Learning Framework

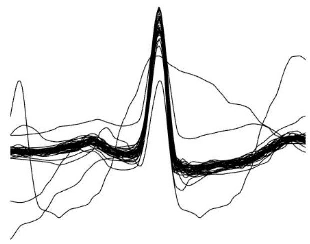
The ECG classifier was implemented through a cascade of binary classifiers. Each classifier recognizes one and only one of three possible labels (A = arrhythmic, O = others, T = Noisy ECG) for an ECG record. Then a cross check of the results give us the final label.

All the algorithm was develop in Python.

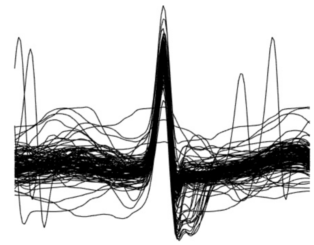
A. *The Binary Classifiers*

The binary classifiers have been implemented through a multi-branches Convolutional Neural Network built with Keras. Each classifier has the same structure that can be seen in figure 7.

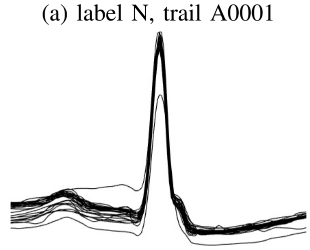
The two branches of the network are identical convolution network. Their outputs are subsequently flattened, passed through a dense layer and merged together. Then the vector pass through 3 dense layers that return a vector with two



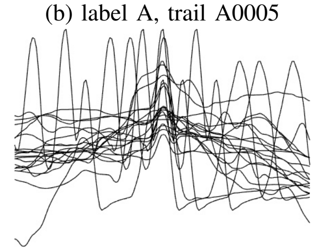
(a) label N, trail A0001



(b) label A, trail A0005

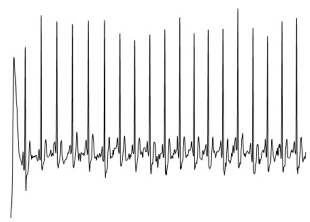


(c) label O, trail A0077

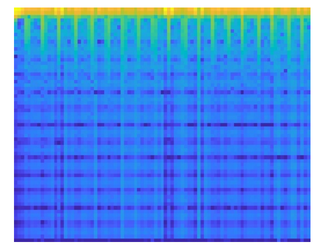


(d) label , trail A0056

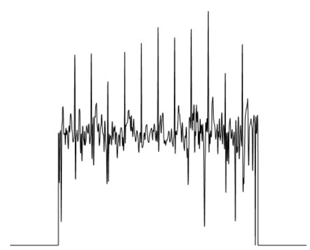
Fig. 5: Some examples of Dataset A.



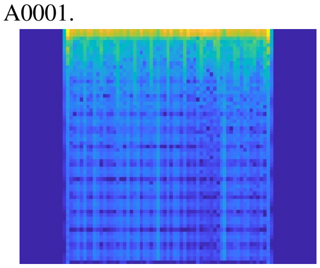
(a) Central window of trial A0001



(b) Spectral representation of the selected signal window of trial A0001.



(c) Trial A0167 is shorted than $15\ s$, so it is padded to create the signal window.



(d) Spectral representation of the selected signal window of trial A0167.

Fig. 6: Some examples of Dataset B.

elements. The first element indicates the probability that the ECG does not belong to the class instead the second element indicates the probability that the ECG belongs to the class.

There are in total 5 convolution layers, each one with a kernel size of 5. The number of the filters of each layer are respectively 8, 16, 32 and 64. The activation function is the *ReLU*. Between each convolution layer and the respective activation there is a batch normalization layer to speed up the training. The dense layer in each branch has a dimension of 500 neurons with *ReLU* activation function. The dimensions of the last 3 dense layer are respectively 200, 50 and 2 neurons. The activation function is *ReLU* for the first two layer and *softmax* for the last layer. Some dropout layers with probability 0.15 were also added to avoid overfitting.

B. Algorithm Structure

The algorithm is divided in two phases: the first is the binary classification of the ECG of categories A, O, T, using *one-vs-rest* strategy, and the second is the cross-check of the previous results to assign the final label.

During the first phase, each trial passes sequentially through the 3 classifiers. The order of the classifiers was determined by their accuracy during the training of the network. The first is the more accurate and the last is least accurate.

The results of this phase is a [3,1] vector with element that can be 0 or 1. Each position in the vector corresponds to one of the three category of the classifier (i.e. the first element corresponds to T, the second to A and the last to O). The value 1 means that ECG record belongs to that class while value 0 is the opposite.

The second phase checks the previous vector to assign the final label with the following criteria:

- If there is only one element with the value 1 means that only one classifier has recognized the signal, so the final label is directly derived.
- If there is no element equals to 1 (i.e. all elements of the vector are 0) means that no classifier recognized the signal. Then the label must be N because is the only option left.
- If there is more than one element with value 1 means that more than one classifier has recognized the signal. In this case the classifier with the higher accuracy wins.

A scheme of the labeling criteria could be seen in Figure 8.

VI. Evaluation and Results

For the training of the networks we used the *ADAM optimizer* with default parameters and the *sparse categorical cross-entropy* loss function. The *batch sizes* was set to 10 and the *number of epochs* was 1 for A and O classifiers and 8 for T classifier.

The results of the classifiers (in the form of confusion matrix) were reported in the table 1.

As a metrics of comparison we used the score of the PhysioNet/CinC Challenge 2017. To evaluate the score we need the following coefficients first:

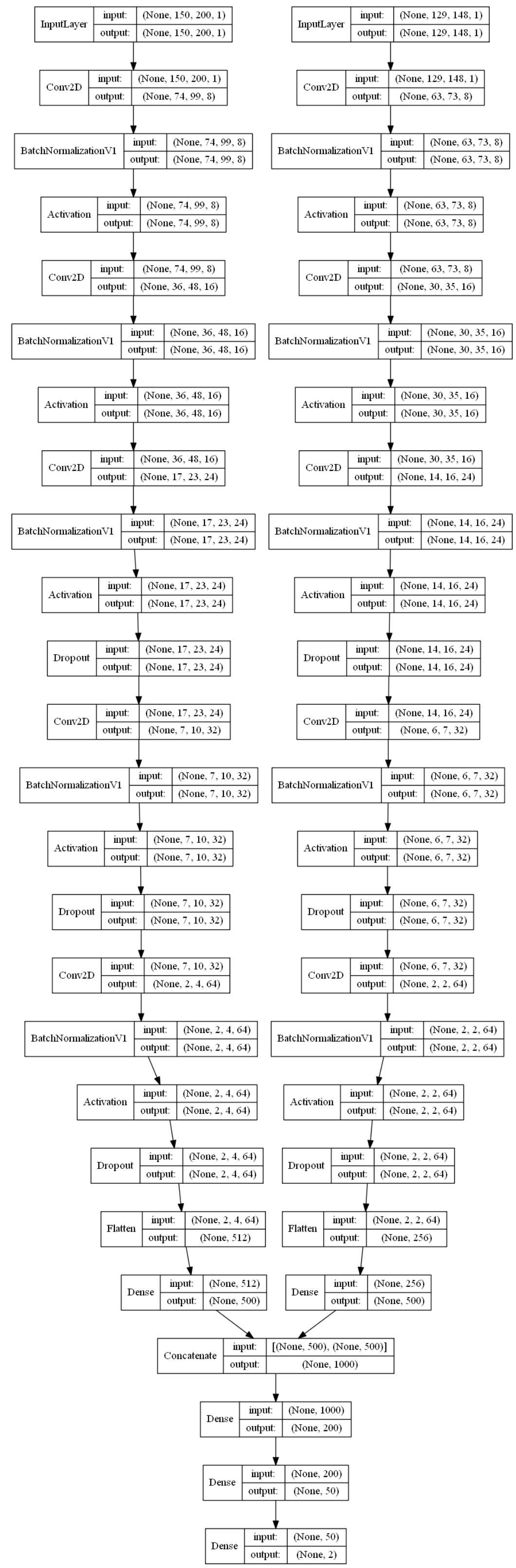


Fig. 7: Representation of a classifier

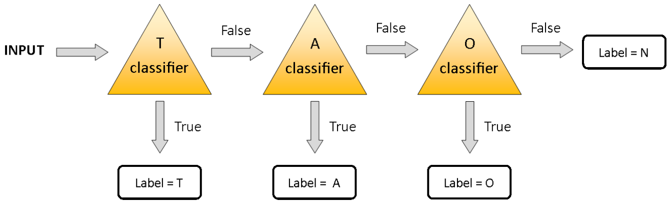


Fig. 8: Scheme of the binary classifier.

	Normal	AF	Others	Noisy
Normal	353 (0.97%)	0	0	10 (0.03%)
AF	41 (0.98%)	0	0	1 (0.02%)
Others	171 (0.98%)	0	0	4 (0.02%)
Noisy	1 (0.01%)	0	0	77 (0.99%)

TABLE 1: Confusion Matrix: columns represent *predicted label*, while the rows represent *true label*.

- Normal rhythm: $F_{1n} = \dfrac{2 * Nn}{\sum n + \sum N} = 0.760$
- AF rhythm: $F_{1a} = \dfrac{2 * Aa}{\sum a + \sum A} = 0$
- Other rhythm: $F_{1o} = \dfrac{2 * Oo}{\sum o + \sum O} = 0$
- Noisy: $F_{1t} = \dfrac{2 * Pp}{\sum p + \sum P} = 0.905$

With this coefficient we can evaluate the final score through the formula:

$$F_1 = \frac{F_{1n} + F_{1a} + F_{1o} + F_{1t}}{4} = 0.416$$

We also report in table 2 the accuracy in the train phase and in test phase of each classifier.

VII. Concluding Remarks

A. Conclusion

Despite the use of datasets with different representations of the signal and the use of different approaches and network architectures, the networks do not improve their capacity of classification. Probably these representations were not characterizing to distinguish the different class of ECG records. A possible solution is to add more information that can facilitate the network to identify the different categories.

The arrhythmia was described as an alteration of the heartbeat rhythm and waveform and in this paper we focused especially on the waveform, rappresented in both time and frequency. A future implementation could include information about heart rhythm like the distances between R peaks, the QRS period, the amplitude of the peaks and so on. A similar approach has been proposed by Sannino et al. [2].

Predicted Classification

Reference Classification		Normal	AF	Other	Noisy	*Total*
	Normal	Nn	Na	No	Np	$\sum N$
	AF	An	Aa	Ao	Ap	$\sum A$
	Other	On	Oa	Oo	Op	$\sum O$
	Noisy	Pn	Pa	Po	Pp	$\sum P$
	Total	$\sum n$	$\sum a$	$\sum o$	$\sum p$	

Fig. 9: Counting rules for the numbers of the variables. (Figure taken from [7])

	AF	Others	Noisy
Train	0.925	0.735	0.984
Test	0.936	0.734	0.976

TABLE 2: Test and train accuracy (The training accuracy is referred to the validation set.)

B. Future Improvement

Future direction for this work could include a better and deeper pre-process analysis to extrapolate more and different families of features from the signal. Subsequently the classifier could be improve with more branches to use this new families of features and improve their performance.

Possible extra features could be directly related to the signal, e.g. QRS period and related information as peak position, distance between peak, spectrograms evaluated with a different method (like the wavelet transform) and with different parameters. Another type of features to be extrapolated could be those obtained through deep learning network (i.e. artificial features).

Others minor improvement could come increasing the level of complexity of the CNNs (i.e. adding more layers and/or filters in each layer) and from the use of more advance activation function like the *LeakyReLU* or the *PReLU*.

C. Difficulties encountered and knowledge acquire

The first difficulty encounters in this work was the impossibility to use a single network to classify the ECG signals. This problem came from the fact that a single network classifier require a very heavy structure with an elevate number of convolutional layers and kernel filters. This leds to network with hundred of millions of parameters that are impossible to train with our current instrumentation.

Instead of abandon completely the use of CNNs we solve this problem with a change of approach. We shifted from the use of a single complicated classifier that extracts by its own the features, to a simpler classifiers that is able to combined the features extracted from different datasets (each one with different representation of the signals).

The low value obtained for F1 is due to the lack of recognition for class A and O. In reality this result is not due to a poor recognition capacity of the respective classifiers but more to the nature of the dataset provided. The dataset is strongly unbalanced because it contains a number of N trials much greater than all the other categories. Consequently, the classifier trains more to recognize what is *not* the corresponding category. This would explain the high number of *false negatives* for categories A, O and P.

The solving of all this problems taught us the importance lateral thinking when a new problems appear (e.g. the use of simple network to avoid hardware constraints). The work on the dateset also made us understand the importance of have different presentation of the information and improve our ability to handle and process the data.

From the analysis of our work it clearly emerge that there are two main approaches to classification problems using CNNs. The first is to used very complicated and deep network with a very basic dataset. The second is to use very simple network that could rely on more representative datasets obtained from a much more complex features extraction previously performed.

REFERENCES

[1] S. H. Jambukia, V. K. Dabhi, and H. B. Prajapati, "Classification of ecg signals using machine learning techniques: A survey," in *2015 International Conference on Advances in Computer Engineering and Applications*, pp. 714–721, IEEE, 2015.

[2] G. Sannino and G. De Pietro, "A deep learning approach for ecg-based heartbeat classification for arrhythmia detection," *Future Generation Computer Systems*, vol. 86, pp. 446–455, 2018.

[3] F. Andreotti, O. Carr, M. A. Pimentel, A. Mahdi, and M. De Vos, "Comparing feature-based classifiers and convolutional neural networks to detect arrhythmia from short segments of ecg," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[4] M. Bettoni and M. Zimmermann, "Autonomic tone variations before the onset of paroxysmal atrial fibrillation," *Circulation*, vol. 105, no. 23, pp. 2753–2759, 2002.

[5] F. Censi, I. Corazza, E. Reggiani, G. Calcagnini, E. Mattei, M. Triventi, and G. Boriani, "P-wave variability and atrial fibrillation," *Scientific reports*, vol. 6, p. 26799, 2016.

[6] Y. Bengio *et al.*, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.

[7] "Af classification from a short single lead ecg recording: the physionet/-computing in cardiology challenge 2017." Accessed: 22/07/2019.

[8] M. Zihlmann, D. Perekrestenko, and M. Tschannen, "Convolutional recurrent neural networks for electrocardiogram classification," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[9] C. Liu, Q. Li, P. B. Suresh, A. Vest, and G. D. Clifford, "Multi-source features and support vector machine for heart rhythm classification," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[10] J. Pan and W. J. Tompkins, "A real-time qrs detection algorithm," *IEEE Trans. Biomed. Eng*, vol. 32, no. 3, pp. 230–236, 1985.

[11] S. Ghiasi, M. Abdollahpur, N. Madani, K. Kiani, and A. Ghaffari, "Atrial fibrillation detection using feature based algorithm and deep convolutional neural network," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[12] "peakdet: Peak detection using matlab." http://billauer.co.il/peakdet.html. Accessed: 22/07/2019.

[13] J. A. Behar, A. A. Rosenberg, Y. Yaniv, and J. Oster, "Rhythm and quality classification from short ecgs recorded using a mobile device," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.