Atrial Fibrillation Detection by Multi-scale Convolutional Neural Networks

Zhenjie Yao*, Zhiyong Zhu*, Yixin Chen†

*Beijing advanced innovation center for future internet technology, Beijing University of Technology, Beijing, China
Email: yaozhenjie@gmail.com,zhiyong.zhu@bjut.edu.cn
†Department of Computer Science and Engineering, Washington University in St.Louis, USA
Email: chen@cse.wustl.edu

Abstract—Atrial Fibrillation (AF) is the most common chronic arrhythmia. Effective detection of the AF would avoid serious consequences like stroke. Conventional AF detection methods need heuristic or hand-craft feature extraction. In this paper, a deep neural network named multi-scale convolutional neural networks (MCNN) based AF detector is proposed. Instant heart rate sequence is extracted from ECG signal, then an end-to-end MCNN detects AF with the instant heart rate sequence as input and detection result as output. The algorithm was tested on both public and private datasets. On the public dataset, with the sensitivity achieved being 0.9822, the corresponding specificity is 0.9811, and the overall accuracy is 0.9818. The area under an ROC curve is as high as 0.9962, compared to the AUC of the best conventional method is 0.9947. Comparison shows that the MCNN based AF detector give superior accuracy than conventional methods. Test on private dataset also shows significant improvement.

I. INTRODUCTION

Atrial Fibrillation (AF) is the most common chronic arrhythmia. According to Framingham heart study, lifetime risk of AF is about 25% [1]. When AF occurs, the frequency of atrial activation is of 300 to 600 times per minute, heart rate can reach 100 to 160 beats per minute. Furthermore, the heart beat rhythm disorders. During AF, Atrium failed to contract effectively leads poor cardiac ejection, and thus easy to form thrombosis. The awful outcome of AF is stroke, which can result in death, paralysis, limb movement or sensory disturbances. The risk of stroke in patients with AF is 5 times more than others [2][3].

Generally, AF is significant different from normal heart rhythm on ECG signal [4]. First, P-wave is absent, and there are continuous, irregular f-wave. At the same time, the heart rhythm disorder, which result in irregularity of RR (R peak to R peak) intervals. It is possible to detect most AF automatically from ECG. Effective detection of AF leads to timely intervention, which would avoid disease progression to serious consequences.

In the last few decades, many researchers tried to detect AF automatically. Lots of effective works has been done. According to the methods adopted, AF detection can be divided into the following categories. The first category is RR interval based algorithm. R waves are identified from the ECG signal, then RR interval is derived. We can detect AF by identify the irregularity of heat pace. Moody and Mark modeled the RR interval as Mokovk process, likelihood of observing pairs of RR intervals was calculated to identify AF [5]. Logan et. al. detected AF according to the variance of RR intervals [6]. Linker determined an instantaneous heart rate for each of the heart beat intervals, and determined a non-linear value that represents the variability of the instantaneous heart rates [7]. Tateno and Glass detected AF by Kolmogorov Smirnov test of RR intervals [8]. Cerutti et.al. found significant difference between AF and normal sinus rhythm, in parameters extracted through a time-variant identification of an autoregressive model and non-linear measurements based on the corrected conditional entropy [9]. Sakar et. al. proposed a series of features derived from the Lorenz distribution of RR intervals to detect AF [10]. Later, a series of works were based on sample entropy and its variants. Colloca et. al. extracted features mentioned in [7], [10] and [11] etc., and recognized AF by SVM [12]. Lee et. al. introduced time-varying coherence functions, which describes the similarity between adjacent segments, to distinguish AF from normal sinus rhythm. Zhou et. al. calculated sample entropy in symbolic dynamic space to identify AF, which showed outstanding detection performance [13]. In 2015, they improved the method and got the best detection performance until now [14]. Petrenase et. al. used fusion of irregularity of RR intervals and bigeminy suppression for AF detection, which result in a very low complexity and high efficient AF detector, only 8 beats is needed for detection [15].

The second category of AF detector is waveform analysis based method and hybrid method. ECG waveform analysis need to determine the location of every parts of ECG, or the frequency constituents. As for hybrid method, it means taking both RR interval and waveform into consideration. Slocum detected AF by waveform analysis, distribution of power and P-wave absence were identified as the evidence of AF [16]. Schmidt et. al. combine irregularity of RR intervals and absence of P-wave for AF detection [17]. Babaiezadeh et. al. calculated a P-R interval variability, a P wave morphology similarity measure and a R-R Markov score for AF detection [18]. Couceiro detected AF based on the analysis of the three main physiological characteristics of AF, P wave absence, heart rate irregularity and atrial activity [19]. Lárburu gave a good survey on AF detection method before 2010. They also compared these methods systematically, which show that R-R
interval based methods outperform waveform based method and hybrid methods comprehensively [20]. This conclusion is reasonable. R peak is most prominent feature, it is robust to noise. However, waveform features such as f-wave and P-wave is not as robust as R peak, they are likely to be impacted by noise and other disturbances. So we tend to take the RR interval based AF detection method.

Recently, deep learning has got great success in image recognition and speech recognition, which made the system performance improve significantly [21][22]. A deep neural network is capable of extracting features from the original input data, it is robust to variances like translation, rotation, and noise. The MCNN (Multi-scale Convolutional Neural Networks) is a deep neural network that designed for time series classification. It showed impressive advantages compared to other methods in time series classification [23]. Inspired by these works, we proposed an MCNN based RR interval analysis method to detect AF from ECG automatically. Experimental results show that the proposed method is capable of detecting AF effectively.

The remainder of this paper is organized as follows. Section II presents our MCNN based RR interval analysis algorithm for AF detection in detail. Section III covers the experimental results on both public and private datasets and its discussion. Finally, in Section IV, the conclusion of this study is drawn and further work is prospected.

II. MCNN FOR AF DETECTION

The detection method is straightforward. Instant heart rate is derived from ECG signal by preprocessing and transformation. Instant heart rate is feed into the MCNN segmentation by segmentation for AF detection. The MCNN is an end-to-end classifier with instant heart rate as input and detection result as output.

A. Preprocessing

First, the R locations should be extracted from ECG signal. You can skip this step if you can read R location from the corresponding annotations. Most public ECG datasets have corresponding annotations by clinicians. RR intervals is easily derived from R locations. Then Instant Heart Rate (IHR) is calculated by:

$$IHR_i = 60 \times f / RRI_i$$

in which $IHR_i$ is the $i$th IHR, $f$ is the sample rate of ECG signal, $RRI_i$ is the $i$th RR interval. For example, if $f$ is 250 Hz, the $i$th RRI is 200, then the IHR is 75 beat per minute.

After the preprocessing and transformation, the ECG signal is transformed into an IHR sequence. Considering most AF detection method need 128 beats, in this AF detection application, we take 127 IHR samples into consideration. For each IHR, The forward 63 IHRs, itself and the backward 63 IHRs format an input IHR subsequence of the MCNN. The MCNN is a binary classifier, whose output is whether it is AF or not.

B. MCNN

MCNN is an end-to-end deep neural network model, as other deep neural network, it incorporates feature extraction and classification in a single framework. The learnable convolutional layers automatically extracts features at different position and in different scales, leading to superior feature representation and better accuracy performance on time series classification.

Figure 1 presents the architecture of the MCNN. The IHR subsequence with length of 127 is the input time series of MCNN. As shown in the figure, the MCNN framework has three sequential stages: transformation stage, local convolution stage, and full convolution stage. Details of each stage is given below:

- **Transformation stage.** Time series has its intrinsic feature in different scale. Long term (also known as low frequency) reflect the overall trends and short term (also known as high frequency) indicate local changes. Generally, High frequency tends to include noises and perturbations. To capture information in different scales and avoid the influence of noises and perturbations, the transformation stage applies various transformations on the input time series. In our implementation, there are 3 types of transformation.

  The first type is identity mapping, which keep the input IHR sequence unchanged, so the length of the identity mapping output is 127. The second is smoothing the original input by low pass filter. A low frequency filter can reduce the variance of time series, which reduce the influence of high-frequency perturbations and random noises. In particular, we employ moving average to smooth the IHR sequence. Mathematically, it is

  $$IHR_{i}^{SM} = \frac{IHR_{i-t} + IHR_{i-t+1} + \ldots + IHR_{i+l}}{2 \ast l + 1},$$

  where $2 \ast l + 1$ is window length of the low frequency filter. We take $l = 2$ and $l = 5$, corresponding output is of length 123 and 117, respectively. In fact, the filter is not limited to low pass filter. All kinds of filter maybe useful.

  The third is down-sampling transformations, which pick one from every k (down-sample factor) samples in the time domain. Mathematically, it is

  $$IHR_{i}^{DS} = IHR_{1+k_1}, i = 0, 1, \ldots, \lfloor \frac{n-1}{k} \rfloor,$$

  in which $IHR_{i}^{DS}$ is the down-sampled sequence, $k$ is the down-sample factor, $IHR_i$ is the $i$th sample in IHR sequence, $n$ is the length of the original input IHR sequence. In the AF detection application, we get only one down-sample sequence with $k = 3$, the length of the output is 43.

  Each transformation result is called a branch, as it is a branch input to the following local convolutional stage.
It is worth noting that transformation is not limited to the mentioned 3 types, other transformations may also work.

- **Local convolution stage.** As described above, after the transformation stage, we obtain multiple time series with different lengths from a single input sequence. In the local convolution stage, we use several convolutional layers to extract the features for each branch independently. All the local convolution filters are with the same window size 8. It is worth noting that with the same window size, convolution on shorter signal means larger local receptive field, while on longer signal means smaller local receptive field. Same filter length on different branch means feature extraction across different scales.

As conventional deep neural networks, all the outputs will pass through a nonlinear down sampling, name max pooling, which can reduce the size of the data, and make the neural network robust to spatial shifting and rotation. Instead of adopting a constant pooling factor, the pooling size for each branch is different. In our implementation, no matter how long is the branch, the length of its output is 12. For example, if the length of a branch is 123, we select pooling factor as 10, and the length of outcome sequence is 12.

- **Full convolution stage.** After extracting feature maps from multiple branches, all these features are concatenated into one sequence, and fed into a full convolutional layers (each followed by max pooling), as well as a fully connected layer followed by a softmax transformation. The output of MCNN will be the predicted distribution of each possible label for the input time series. We could derive final result by simply taking the label with maximal possibility.

As shown in Figure 1, the MCNN is an end-to-end system. No heuristic or hand-craft feature extraction is needed. When labeled training dataset is available, all the parameters could be trained jointly through back propagation. In the application of AF detection, the MCNN served as a binary classifier. AF is positive example, and other heart rhythms (including normal rhythm and other arrhythmia) are negative examples.

### III. Experimental Results

Three datasets are adopted to evaluate the MCNN AF detector, including the Long-Term AF Database (LTAFDB), the MIT-BIH AF Database (AFDB), and the Holter ECG dataset (HEDB). LTAFDB, and AFDB are public datasets, could be accessible from [24]. HEED is a private dataset, which is constitute of 499 Holter ECG signals, each is about 24 hours long. The summary of all the datasets are listed in Table I. Informations include the sample frequency, total number of beats, number of AF beats, number of records and average duration of each records. For comparison, we follow the way of dataset division in [14],the LTAFDB database is used as the training set to determine the model parameters, while AFDB and HEED are used as the testing sets. Besides of this test, we also evaluate the proposed algorithm on HEED with half data as training set and the other half as testing set.

The sensitivity, specificity and accuracy are calculated to measure the performance of the proposed method for AF detection. They are defined as:

\[
\text{sensitivity} = \frac{TP}{TP + FN},
\]

\[
\text{specificity} = \frac{TN}{TN + FP},
\]

\[
\text{Accuracy} = \frac{(TP + TN)}{(TP + FN + TN + FP)},
\]

in which TP (True Positive) is the number of AF beats that are recognized as AF; FN (False Negative) is the number of AF beats that are recognized as not AF; TN (True Negative) is number of not AF beats that are recognized as not AF beats; FP (False Positive) is number of not AF beats that are recognized as AF beats. In general, a good detection method will maximize the all the 3 performance metrics.

Table II lists the tests results on AFDB together with four existing methods that tested on the same dataset, with LTAFDB as training dataset. To our knowledge, the four methods give better results in the existing methods. Zhou et. al. [14] give the best detection performance compared to other existing works, the accuracy is 97.99%. The proposed MCNN based AF detection method improved the detection performance significantly, the accuracy is 98.18%, while the sensitivity is 98.22% and the specificity is 98.11%. Despite the accuracy only increased 0.19%, but we essentially reduced the error of about 10%. So we conclude that, on AFDB dataset, the MCNN based AF detection outperforms the best of the existing methods.

Since symbolic dynamics and Shannon ENtropy (SEN) [14] shows best detection performance compared to other existing works, we realized the method for extensive comparison.

In statistics, a receiver operating characteristic (ROC) curve is a graphical plot that illustrates the performance of a detection system as its discrimination threshold is varied. The curve is created by plotting the sensitivity (also known as Detection Rate, DR) against the False Alarm Rate (FPR) at various threshold settings. FAR is defined as:

\[
FAR = 1 - \text{specificity}.
\]

Figure 2 shows the ROC curves obtained by MCNN and SEN on AFDB dataset, respectively. Most parts of the two curves are similar. To enlarge the detail of the ROC curves, only the top-left part is cropped. Both curves are interpolation results of 21 actual test points. Each test point contains a pair of DR and FAR values. The test points of MCNN are obtained by modifying the threshold on the output of softmax layer, each actual test point is corresponding to a threshold. Conventional softmax layer is corresponding to threshold of 0.5. As for SEN, tuning the threshold of the final entropy leads to the ROC curve. As shown in the figure, when the FAR is smaller than 0.02, corresponding to the very left part of the figure, the two curves are so similar that they almost coincide with each other. In the other parts of the curves (FAR>0.02), MCNN outperform SEN consistently.
The AUC is a measurement of the detection accuracy, which is defined as the ratio of the area under the ROC curve to the area of the whole figure. The larger the AUC is, the better performance the detector achieves. For the AF detection problem, the AUC of MCNN is 0.9962, while the AUC of SEN is 0.9947, which implies that the proposed MCNN based AF detection method is more effective than SEN based method.

Table III lists the test results on HEDB. As HEDB is private dataset, no test result exists in other literatures. As shown in the table, while the sensitivity is 95.87%, the specificity is 93.90%, the accuracy is 94.93%. The proposed MCNN based AF detection method failed to improve the detection performance, the accuracy is 94.32%, while the sensitivity is 95.57% and the specificity is 92.89%.

We infer that the failure on HEDB is caused by difference between the training dataset (LTAFDB) and test dataset (HEDB). Both LTAFDB and AFDB are from Physionet, all the ECGs are collected from Americans; but HEDB are collected in China. Because ethnic characteristics, there are significant difference between the datasets. We check the mean heart rate of all the 3 dataset to valid the guess. The mean heart rate of LTAFDB is 91.36 beats per minute. The mean heart rate of AFDB is 88.07 beats per minute. The mean heart rate of HEDB is 83.09 beats per minute. We see that compared to
Fig. 2. ROC curves on AFDB.

TABLE III

<table>
<thead>
<tr>
<th>Year</th>
<th>Sensitivity(%)</th>
<th>Specificity(%)</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhou et. al. [14]</td>
<td>95.87</td>
<td>93.90</td>
<td>94.93</td>
</tr>
<tr>
<td>Proposed</td>
<td>95.57</td>
<td>92.89</td>
<td>94.32</td>
</tr>
</tbody>
</table>

TABLE IV

<table>
<thead>
<tr>
<th>Year</th>
<th>Sensitivity(%)</th>
<th>Specificity(%)</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhou et. al. [14]</td>
<td>94.98</td>
<td>95.20</td>
<td>95.09</td>
</tr>
<tr>
<td>Proposed</td>
<td>96.32</td>
<td>95.58</td>
<td>95.98</td>
</tr>
</tbody>
</table>

the difference between LTAFDB and AFDB, the difference between LTAFDB and HEDB is much more significant. The difference between training dataset and test dataset is the main cause of the failure on HEDB.

Furthermore, we divide the HEDB into two datasets. It is worth noting that, the separation is person-independent. The ECG of each person will only appear in one subdataset. Cross validation is taken to evaluate the MCNN based AF detector, the two subdataset take turns to serve as training dataset. For the motivation of justice, the parameters of SEN are also optimized for HEDB. Results are shown in Table IV. For the optimized SEN method, while the sensitivity is 94.98% and the specificity is 95.20%, the accuracy is 95.09%. Only a little improvement is achieved. The proposed MCNN based AF detection method improve the detection performance, the accuracy is 95.98%, while the sensitivity is 96.32% and the specificity is 95.58%. We see that while both training and testing dataset are from HEDB dataset. The MCNN based AF detection outperforms SEN based method.

IV. CONCLUSIONS AND FURTHER WORKS

This paper investigated AF detector based on MCNN. First, instant heart rate sequence is extracted from ECG signal. Then an end-to-end MCNN was trained to differentiate the
sequence is AF or not. The test results on both public and private dataset show that the proposed method is effective. On the public dataset, with the sensitivity achieved being 0.9822, the corresponding specificity is 0.9811, and the overall accuracy is 0.9818. The area under an ROC curve is as high as 0.9962, compared to AUC of 0.9947 for the best result until now. As for the private dataset, the significant difference between training dataset and test dataset is the main cause of the failure on HEDB. If take half of the dataset as training dataset and half as test dataset, the MCNN based AF detector outperform traditional SEN based AF detector. Comparison show that the MCNN based AF detector give higher accuracy than conventional methods.

The testing results on HEDB show that the deep learning method is more sensitive to the difference between training and testing datasets. Such difference usually exists in real life scenarios. Transfer learning is branch of machine learning that specially developed for solving this problem. We infer that transfer deep learning model could accomplish the work.

Although the performance of the proposed method is good, the AF detection problem is far from being solved. There are still quite a lot of problems worth exploring. According to the literatures and our experience, bigeminy, trigeminy, and sinus arrhythmia are easy to be mistaken for AF, which would lead to false alarm. It is hard to distinguish AF and part of these arrhythmias. More attention should be paid to these problems.

Another concern is that compared to conventional AF detection method, deep learning is computationally intensive. It is difficult to calculate in real-time, especially for wearable devices with limited power supply and limited computation capacity. Generally, 128 heartbeats usually last 1 minute and a half, most ECG last only 10 seconds. We would try to detect AF in 10 seconds in our further work.

MCNN show power on AF detection, deeper model may improve the result. Besides modifying the MCNN, another important kind of deep learning model should also work on AF detection. It is Recurrent Neural Networks (RNN), which is an important kind of deep learning model.

REFERENCES

ACKNOWLEDGEMENT

This work is in part supported by US NSF grants IIS-1343896, DBI-1356669, and IIS-1526012.