ECG ARRYTHMIAS RECOGNITION SYSTEM BASED ON FUSION OF PROBABILISTIC NEURAL EXPERT

Hendel mounia1, Benyettou Abdelkader1, Hendel Fatiha2, Khelil Hiba1
1 SIMPA Laboratory, Department of Computer Science, Faculty of Science, University of Science Technology of Oran, Mohammed Boudiaf, USTO, Algeria
2 LARESI Laboratory, Department of Electronic, Faculty of Electric, University of Science Technology of Oran, Mohammed Boudiaf, USTO, Algeria
mounia_90@hotmail.com

ABSTRACT

In this paper, an electrocardiogram (ECG) beat classification system is proposed to discriminate four ECG beat types. This system is based on a fusion of two probabilistic neural networks; our main objective is to minimize the risk of error diagnosis. The first network is RBFNN (Radial Basis Functions Neural Network), it receives 11 temporal parameters and morphological characterizing an ECG beat, and these parameters are mainly determined using wavelet transform. The second network is a BPNN (Back Propagation Neural Network), it receives directly 256 input amplitudes of samples of an ECG beat. Four types of arrhythmias are considered in this study namely, premature ventricular contraction ventricular (V), Atrial premature contraction (A), Right Bundle Branch Block (RBBB) and Left Bundle Branch Bloc (LBBB), in addition to normal beat (N). Our system has been validated on recordings taken from the MIT-BIH ECG data and achieves classification accuracy higher than 98%.

Keywords: Electrocardiogram (ECG) beats, MIT-BIH database, Probabilistic neural networks, data fusion.

1 INTRODUCTION

Electrocardiography is an important tool in diagnosing the condition of the heart. The electrocardiogram (ECG) is the record of variation of bioelectric potential with respect to time as the human heart beats. It provides valuable information about the functional aspects of the heart and cardiovascular system. Early detection of heart diseases /abnormalities can prolong life and enhance the quality of living through appropriate treatment. Therefore, numerous research and work analyzing the ECG signals have been reported, this methods are based on a number of features and different classification algorithms. The features include time domain [1-5], frequency domain [6-8], statistical measure [9],or represented as time-frequency domain [10-13].The classification techniques include discriminate analysis [3], back propagation neural networks [14] and self-organizing feature maps using neural networks [15], probabilistic neural networks [16] support vector machines [10] and independent component analysis (ICA) [17].

In this paper, we use two probabilistic neural networks, and we combine two architectures in order to improve diagnostic accuracy. Indeed, the fact that having more than one primary classifier in the system, should improve its performances. In general, different classifiers work on different principles and tend to make independent errors. By combining the results of several such systems, we might assume that the errors will annul and that the overall performances of the system will improve.

As primary classifiers, we have choose the RBFNN and BPNN which are two universal approximate neural networks. The first neural network receives as input data, 11 temporal and morphologic parameters characterizing the ECG beat; for the second network, the same ECG beat is simply introduced as it is as 256 samples. The outputs of the two classifiers are then fused using the Dempster Shafer rule. The main goal of this work is to realize a robust classifier able to identify all the above cited types of Heartbeats by using neural networks on a Bayesian framework. The figure 1 presents the global scheme of the proposed system.

The rest of this article is structured as follows. In session II, we present steps and algorithms used to extract beat signal parameters from MIT-BIH data base. In session III, we present the methodology adopted for the two classifiers. In section 4 we display the results obtained for a set of real ECG signals from the MIT-BIH database and finally in section 5 we present some conclusions.
2 PARAMETRES EXTRACTION

2.1 ECG data base
The data for this study were obtained from the MIT-BIH Arrhythmia database. The collection of annotated ECG signals in this database has been used in a number of studies. The database contains records obtained from 48 subjects studied by the Arrhythmia Laboratory of the Beth Israel Hospital between 1975 and 1979. Signals in each record are sampled at 360 Hz and are approximately 30 min in length. For each record, there is a corresponding annotation file, created by qualified cardiologists, that identifies the category of each beat.

2.2 preprocessing
ECG signals can be contaminated with several types of noise, such as power line interference, electromyographic noise and baseline wandering, which will imply the displacement of the isoelectric line of the ECG. To remove the nondesired components of signal ECG, we thus applied to the recordings a pretreatment [18] which consists of two filtering passes low (1) followed by a filtering passes high (2). The transfer transfer functions of the two filters are:

\[ L(Z) = \frac{1 - 2Z^6 + Z^{12}}{36 - Z^1 + Z^2} \]  
\[ H(Z) = \frac{-1/32 + Z^{16} - Z^{17} + 1/32Z^{-32}}{1 - Z^{-1}} \]

2.3 Extraction of the input parameters for the RBFNN
The QRS complex is the most characteristic waveform of an ECG signal and its width has been a diagnostic criterion of cardiac arrhythmia. In order to determine the complex QRS, we have used the Pan and Tompkins method [2], with a reasonable complexity and moderate computing time. The final goal of the method Pan and Tompkins is the localization of the peak R for calculates heart rate, thus it helps us to detect the waves Q and S for measured lasted of complex QRS. The various steps which constitute it are:

- Calculate the first and second derivate \( y_1 \) and \( y_2 \) of ECG signal \( x(n) \):
  \[ y_1 = x(n + 1) - x(n - 1) \]  
  \[ y_2 = x(n + 2) - 2x(n) + x(n - 2) \]  

- Smoothing of the two derivatives: Smoothing is obtained by averaging over 3 consecutive samples, we set:
  - \( y_{liss1} \): Smoothing of the first derivative
  - \( y_{liss2} \): Smoothing of the second derivative

- Summation
  \[ y(n) = y_{liss1}^2(n) + y_{liss2}^2(n) \]

The obtained signal is passed through a window of integration of \( N \) points:

\[ z(n) = \left( \frac{1}{N} \right) \sum_{i=1}^{N} x(n - (N - i)) \]

- Calculate the positions of QRS complex by a threshold set at 30% of the maximum value of \( z \).

Fig.2. detection of the QRS complex.

We have used the same method for detecting the
waves T and P; it is based on wavelet transform of the heart signal. After several tests, we have choose the best coefficient N=20 for localising this waves.

Once the various waves of a heartbeat are located, we can then calculate 11 parameters namely: RR interval, QRS complex during, R peak amplitude, S wave amplitude, presence or absence of the Q wave, during and amplitude of wave P, PR interval. These parameters are normalised and represent the input data for the RBFNN classifier. A normalization process is necessary to standardize all the features to the same level. The hyperbolic tangent sigmoid function \[ \tanh \] is used to transform each feature to the same range of \([-1, +1]\]. The mean and the standard deviation of each component in the feature vectors are calculated from the training dataset and used throughout the experiments. From the two former steps ten.

4.2 The BPNN input data
The input data for the BPNN are the beat heart which is simply presented as it is, to the classifier as vectors of 256 samples centered into the peak R, so that we are sure that all beat waves will be included and the morphology of the beat is preserved.

5.1 data base preparation
We have created from MIT-BIH database two data sets, one for training the two classifiers, and the second for the testing phase. The selected beats are given in the following table:

<table>
<thead>
<tr>
<th>Type</th>
<th>Patients MIT-BIH</th>
<th>Training data</th>
<th>Test data</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>100,103,105,108, 112,113,114,115,117</td>
<td>500</td>
<td>2500</td>
</tr>
<tr>
<td></td>
<td>109,111,207,214</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LBBB</td>
<td>118,124,212,231,232</td>
<td>500</td>
<td>1500</td>
</tr>
<tr>
<td>RBBB</td>
<td>106,200,119, 214,203 ,208</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>V</td>
<td>100,209 ,118,202, 200,213,220</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>A</td>
<td>124,212,231,232</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>2500</td>
<td>6000</td>
<td></td>
</tr>
</tbody>
</table>

3 CLASSIFICATION
The Bayesian statistic decision (TDSB) \[19\] allows optimal decision-making by minimizing a test error. The Bayesian decision rule can be given as following:

\[
P(Y_k) \cdot P(X|Y_k) > P(Y_j) \cdot P(X|Y_j) \quad \forall k \neq j
\]

However, in real problems the probabilities \(P(Y_k)\) and \(P(X|Y_k)\) are generally unknown. Therefore, we need to a powerful tool in order to estimate the exact distribution values.

On the other hand, neural network are a universal approximate , they can estimate any non-linear function from a set of samples. In particular, they can be used to determine the membership probability to different classes: \(P(Y_k / X)\).

Assume that we have the training set D, consisting of N input-output pairs:

\[
D = \{X^n, Y^n \}_{n=1,2,...,N}
\]

Where X is an input vector consisting of L elements, every input vector corresponds to a heartbeat, and Y is the corresponding class label.
consisting of $K$ classes. The goal is to use an ANN to model the input-output relation ($Y = k/X$). In our case we have five classes: $(k = k_1, k_2, k_3, k_4, k_5)$, where the first four classes correspond to the four types of arrhythmia. Any one of them takes a values (0) for the normal, and (1) for the arrhythmic heart beat. The five class corresponds to the normal case, and takes a value (1) for the normal beat and (0) in the case of arrhythmic one. To realize a logistic regression model based on a Bayesian method, we estimated the class probability for the given input by: $P(Y = k/X)$.

Our probabilistic neural network classifier’s configurations are shown schematically in figures 6 and 7.

### 3.1 BPNN classifier

The architecture used to develop the first classifier is a four-layer feed-forward neural network (fig.6).

- The first layer is the input layer that has 256 samples of a beat centred on the pick R as inputs.
- The second and the third layers, also called the hidden layers, have respectively 54 and 32 neurons. To determine the optimal number of hidden neurones, we realised several trainings for different architectures. A validation data base enabled us to choose among all these architectures that witch gives in general the best rates.
- The output layer has five neurons, which is equal to the number of ECG beat types to be classified, and are regarded as membership probability of classes. So we have used the log sigmoid transfer function in this layer to have an output value between 0 and 1.

In this architecture, the first and the second layer carried out a kind of projection of the 256 components of the input vector on a space with 54 dimensions, as would do a principal component analysis (PCA) [10]. The classification is made by the third and last layer.

The network has been trained by using backpropagation algorithm with moderate values of learning rate and momentum. The weights are updated for every training vector, and the termination condition is that the sum square error reaches a minimum value. The connection weights are randomly assigned at the beginning and progressively modified to reduce the overall system error. The learning database is composed of 2500 beats examples divided to five types of classes (Table 1).

### 3.2 RBFNN Classifier

The goal of supervised RBF networks is to approximate the desired behaviour by a collection of functions, called kernel. The RBF network is constituted of three layers only:
The input layer which relays the inputs without distortion and the neurons number is equal to the vector size (11 parameters in our case).

- The hidden layer, is composed of kernel function Gaussian type with center $C_i$ and receiver field $\sigma_i$. The output of the hidden neurone is given by:
  \[
  F_i(X) = \sum_{j=1}^{n} \frac{(X_j - C_{ij})^2}{\sigma_{ij}^2} \]

The response of Gaussian function is locale; its maximum value was in center $C_i$, and decreases in a manner to fairly monotone to the distance.

- The output layer: the number of neurons is equal to the number of classes (in our case there is 5 classes). The output value is calculated by:
  \[
  F_k(X) = w_{k0} + \sum_{i=1}^{H} w_{ki} \cdot \Phi(X) \]

$F_k(X)$ represents the posterior probability $P(Y = k \mid X)$ of the class $k$ knowing that $X$ is the input vector; $w_{ki}$ is the synaptic weights between the hidden and output layer, with $1 \leq i \leq H$ and $H$ is the number of neurones in hidden layer (80 in our case).

The RBF learning is done by OLS algorithm "Orthogonal Least Square". The algorithm proceeded in two steps. First, it separate linearly between input and hidden layer; it creates hidden neurons automatically by applying the orthogonalisation of Gram Schmidt which eliminates duplication of information. He subsequently directed learning between the hidden and output layer, by calculating the synaptic weight based on the method of least squares.

4 RESULTS

ECG records with normal beats and different types of arrhythmias are selected from the MIT-BIH arrhythmia database for the analysis. The trained network has been tested with the testing vectors which are not taking part in the training process. The Accuracy of an ECG classifier is defined as the ratio of the number of beats correctly classified to the total number of beats tested. The correct classification rate per class and global rate are summarized in Table 2.

Table 2: The Learning and test classification rate (%)

<table>
<thead>
<tr>
<th></th>
<th>BPNN Learning</th>
<th>BPNN Test</th>
<th>RBFNN Learning</th>
<th>RBFNN Test</th>
<th>FUSION (RBFNN+BPNN) Learning</th>
<th>FUSION (RBFNN+BPNN) Test</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Correct classiﬁcation (%)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>LBBB</td>
<td>99</td>
<td>96.6</td>
<td>100</td>
<td>98.6</td>
<td>98.6</td>
<td>96.2</td>
</tr>
<tr>
<td>RBBB</td>
<td>99</td>
<td>95.8</td>
<td>100</td>
<td>97.6</td>
<td>97.6</td>
<td>95.4</td>
</tr>
<tr>
<td>EV</td>
<td>100</td>
<td>99.6</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>99.6</td>
</tr>
<tr>
<td>EA</td>
<td>100</td>
<td>98.8</td>
<td>100</td>
<td>99.2</td>
<td>99.4</td>
<td>98.8</td>
</tr>
<tr>
<td><strong>Global (%)</strong></td>
<td><strong>99.4</strong></td>
<td><strong>98.16</strong></td>
<td><strong>100</strong></td>
<td><strong>99.08</strong></td>
<td><strong>99.12</strong></td>
<td><strong>98</strong></td>
</tr>
</tbody>
</table>
We considered that a beat is correctly classified by both networks, only if the posteriori probability of a class is higher than 0.75. We also used the fusion probability, i.e. that the output of the fusion unit is simply the product of the probabilities of the two classifiers, a beat is correctly classified by this unit if the probability of class membership is greater than 0.5, so the two classifiers are in agreement.

Table 2 shows that the bad classifications rate for the two classifiers are very low for learning samples; in general the global rate is more than 98%. This rate was slightly decreased after fusion, this is logical, because for few beats, the two networks are not in agreement.

Table 3 shows the fusion matrix between classes using the two networks and also after fusion. We notice that the most confused classes by these modules are the Right and Left Branch Block, because the characteristics of such beats are very similar. We also note that the rejection rate is higher after fusion; in fact, when the two classifiers are in disagreement they can consider beat as rejected (It is better to say I do not know that making a wrong diagnosis).

5 CONCLUSION
In this article we have proposed systems to aid diagnosis of four cardiac arrhythmias which are very common: premature ventricular contraction ventricular (EV), Atrial premature contraction (EA), Right Bundle Branch Block (RBBB) and Left Bundle Branch Block (LBBB), in addition to normal beat (N). Our system combines two modules that work differently:
- For the first one, we begin by localizing QRS, P, T waves and calculating temporal and morphological parameters which characterize a beat using the technique of wavelet transform. A probabilistic RBF neural network is then used to determine the beat type based on its characteristics.
- The second module employs a direct method to extract the parameters which consists in taking 256 samples of the beat centered on peak R. A Bayesian MLP neural network with two layers is then used as well to compress the input vector as to determine its type.

The two modules work in parallel, each one determines the beat class in term of posterior probability, and they are regarded as two different sources each one having an opinion to give. The two module outputs are then fused by using the fusion probability, to reduce at the maximum the errors risk. Our system was validated on real ECG records taken from the MIT-BIH database. The obtained results prove the effectiveness of the suggested method for the diagnosis computer-assisted of the cardiac diseases based on ECG signals.

6 REFERENCES
classification method using Mahalanobis distance.