A Comparison Study on Training Optimization Algorithms in the biLSTM Neural Network for Classification of PCG Signals

Mahmoud Fakhry and Abeer Fathallah Brery

February 16, 2022

A Comparison Study on Training Optimization Algorithms in the biLSTM Neural Network for Classification of PCG Signals

Mahmoud Fakhry
Department of Electrical Engineering
Aswan University
Aswan, Egypt
m.fakhry@aswu.edu.eg

Abeer FathAllah Brery
Department of Electrical Engineering
Aswan University
Aswan, Egypt
abeer.brery@aswu.edu.eg

***Abstract*—A trained neural network classifier is often used to detect cardiac problems by the classification of heart sound signals, also known as phonocardiogram (PCG) signals. The choice of an appropriate training optimization algorithm for such a classification problem, on the other hand, is still being debated. In this paper, we use the bidirectional long short-term memory (biLSTM) network for the classification of sequences of short-time features extracted from labelled PCG signals. The classification performance of four different trained biLSTM models is described in terms of three different optimization algorithms that are used to train the classifier. The elaborated results on testing PCG signals showed that the biLSTM classifier performs better when trained with the stochastic gradient descent with momentum (SGDM) algorithm than when trained with the RMSprop (root mean squared propagation) optimizer or the adaptive moment (ADAM) optimization algorithm. Furthermore, this classification method outperforms a baseline method.**

***Index Terms*—Heart sound, PCG signals, feature classification, biLSTM model, SGDM, RMSprop, ADAM.**

I. Introduction

Automatic diagnosis of disease is focused on the development of robust and dynamic noninvasive systems. As for cardiac disease – one of the leading causes of death across the globe – cardiologists are used to examine the health of the heart by hearing its sound with a medical stethoscope. This diagnosis strategy requires expertise to be learned for many years. In this manner, it was the beginning to think of the automated examination of the health of the heart through computerized investigation of recordings of its sound [1], [2].

Besides, the electrocardiogram (ECG) [3] and the photoplethysmogram (PPG) [4], the phonocardiogram (PCG) [5], the recording of the sounds and murmurs made by heart, can be effectively employed to examine the health of the heart. The ECG and PCG are highly correlated signals, and they are known to contain more information than the PPG signal. The PCG signal, however, enjoys a distinct advantage over the ECG and PPG signals as it records the acoustic properties, which are better suited for the detection of heart abnormality.

The PCG signal of a healthy heart composes of the first heartbeat S_1 and the second heartbeat S_2, and silent time intervals in between of them, referring to the interval that the heart muscle takes to switch from closure to contraction and vice versa. The interval from S_1 to S_2 is known as the systolic region, and the one between S_2 and S_1 is known as the diastolic region. Heart abnormalities are indicated by other audible activities and murmurs that arise in the silent time intervals. Murmur is a noisy cardiac sound that occurs when a heart valve closes, but blood continues to flow. The heartbeat $S1$ is normally a low-frequency, high-intensity signal, whereas the heartbeat $S2$ is a high-frequency, low-intensity signal. Several methods have been presented to detect heart abnormality by classifying PCG signals using trained classification models.

Classification models built based on features extracted from raw signals are more efficient than those based on raw signals. The short-time Fourier transform (STFT) [6], the wavelet transform [7], and the mel frequency cepstral coefficients (MFCCs) [8] are commonly used for feature extraction from PCG signals. In [9], authors carried out a comparative study of the STFT, the wavelet transform, and the time-domain analysis of PCG signals. Their findings suggest comparable spectral and temporal resolution of cardiac acoustical events.

Support vector machine (SVM) is employed for the detection of heart abnormalities, where the wavelet transform coefficients and the linear predictive coding parameters are used as classification features in [10] and [11], respectively. Two feature extraction methods based on curve fitting and fractal dimensions of PCG signals are presented in [12]. Moreover, the authors examined the performance using the k-nearest neighbors (kNN) classifier. In [13], wavelet decomposition, Hilbert transform, homomorphic filtering, and power spectral density (PSD) are exploited to extract features which are classified using the kNN classifier in [13]. Features extracted by applying the principal component analysis and wavelet analysis are classified using the KNN classifier in [14], [15].

Multilayer perceptron (MLP) with one hidden layer and another with two hidden layers were evaluated for the detection of heart abnormality in [16]. Classification features are extracted from the wavelet transform and the STFT of PCG signals. In [17], [18], the feed-forward neural network is used for such task with features extracted from the time, frequency, and time-frequency representations of PCG signals. The wavelet coefficients are employed as features for the classification of PCG signals using the convolutional neural network (CNN) in [19], and the MFCCs in [20]. A combination of time-frequency heat maps and CNN is presented for the identification of cardiovascular disorders in [21]. In [22], four different recurrent neural networks (RNNs) [23] are evaluated separately with the MFCCs for accomplishing such classification task. A combination of two neural networks, i.e., namely the CNN network and the bidirectional long short-term memory network (biLSTM) with a technique to learn visual and time-dependent characteristics of murmurs is explained in [24]. Classification features are obtained based on the spectrogram and the MFCCs of PCG signals. In [25] the nonlinear autoregressive network with exogenous inputs (NARX) is exploited for the diagnosis of heart abnormality with spectral, temporal, and statistical classification features.

It is obvious that artificial neural networks (ANNs) are widely used classification tools, but the selection of an appropriate ANN model is still being questioned. It is challenging to find the best model that could accurately classify the input features and optimize many factors, such as the processing speed, numerical precision, and memory requirements. Such an optimization problem lies in the learning process of ANNs and could be solved by using an appropriate training optimization algorithm. Training a neural network is the problem of minimizing a large-scale cost function. This process is solved using an optimization algorithm that searches through a space of possible values for the neural network weights for a set of weights that results in good performance on the training dataset. A given training optimization algorithm might be suitable for a given problem but might fail in another case.

In this paper, we propose an experimental comparison study on three training optimization algorithms used for training the bidirectional long short-term memory (biLSTM) network for the task of binary classification of PCG signals. These algorithms are the stochastic gradient descent with momentum (SGDM), the root mean squared propagation (RMSprop), and the adaptive moment estimation (ADAM) [26]. This is done using 10 sequences of feature extracted from each PCG signal of selected labeled training signals from the PhysioNet 2016 dataset [27]. Four different models of the biLSTM network are trained and tested using sequences of feature extracted from the selected PCG signals. The classification accuracy of each trained model is reported for the three optimization algorithms.

The rest of this paper is organized as follows. The proposed methodology is explained in Section II. Section III reports the experimental results, and the work is concluded in Section IV.

II. Methodology

One of the most difficult aspects of studying PCG signals is the fact that they are complex, non-linear, and non-stationary. Only within short-time blocks of signal samples are they deemed stationarity. Statistical qualities can be expected to be accurate enough for these blocks in practical applications.

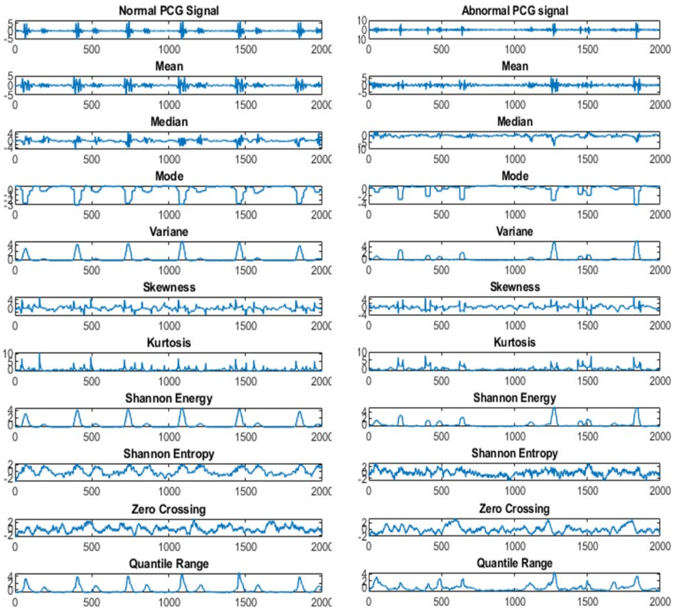

Fig. 1. Short-time feature sequences extracted from two raw PCG signals obtained using a Gaussian window of length 75 ms.

In this paper, a PCG signal is split into overlapped short-time blocks of signal samples and 10 classification features are extracted separately for each block. This is done using the Gaussian sliding symmetric window with a length of 75 ms. This window defines the boundary and contribution of each sample within the block. The Gaussian window is moved slightly along the time representation of the PCG signal with one sample translation step, and the statistical properties of mean, median, mode, variance, skewness, kurtosis, Shannon energy, Shannon entropy, zero-crossing rate, and quantile range are computed within each block [28]. Figure 1 shows normalized sequences of the above features extracted from normal and abnormal PCG signals. Each normalized sequence is with a zero mean and a unity standard deviation.

These ten normalized sequences of features extracted from labelled normal and abnormal PCG signals are used to train, validate and test the biLSTM model exploited as a feature classifier. The proposed methodology is depicted in Figure 2. The methodology starts with signal preprocessing, through the removal of noise and outlayers from each PCG signal, then moves on to the extraction of 10 sequences of features from each preprocessed signal, and finally to the classification of the extracted features for the identification of cardiac anomalies. In the context of PCG signal classification, we directly analyze the influence of choosing a specific optimization algorithm for training the biLSTM model. Among the different algorithms that can be used to train the biLSTM model, we choose the stochastic gradient descent with momentum (SGDM), the root mean squared propagation (RMSprop), and the adaptive moment estimation (ADAM) algorithms.

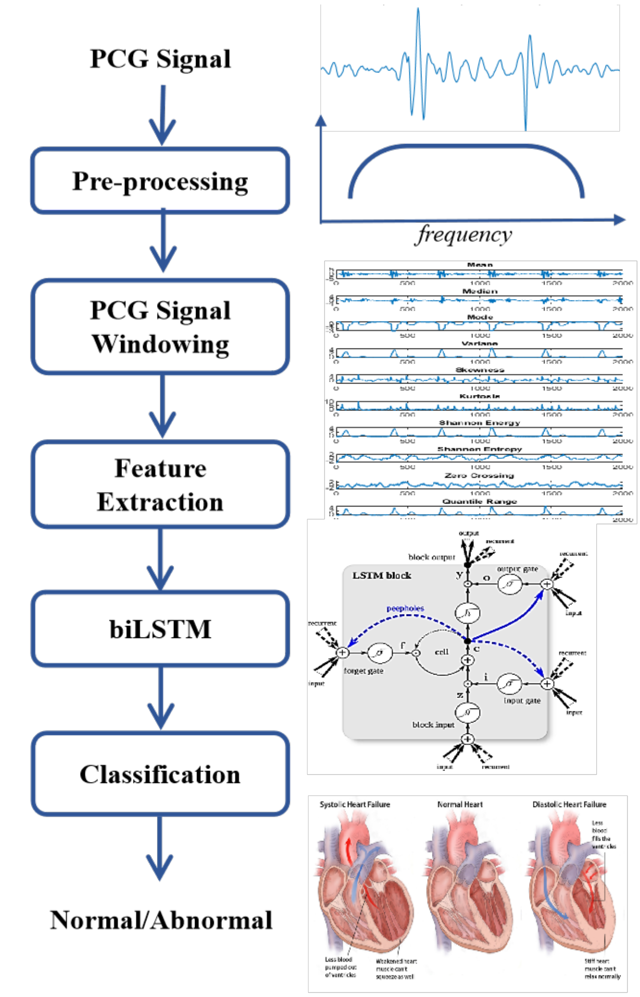

Fig. 2. The flow graph of proposed methodology.

A. *biLSTM model*

RNNs, or recurrent neural networks, can learn arbitrary sequences [23]. RNN models start with an input layer and end with an output layer. Between the input and output layers, there must be at least one recurrent layer. A layer of RNNs is composed of multi-neurons (multi-nodes) and feedback connections. Scalar weights and nonlinear activation functions connect these layers. A PCG signal is represented here by sequences of 10 extracted features. A sequence input layer receives these sequences at the input of the RNN model. After that, the features are passed on to the recurrent layer, which sends its output to the output layer of the RNN model.

1) LSTM unit: Long short-term memory (LSTM) networks are RNNs in which the output of the previous step is used as the input for the following step [29]. RNNs are unable to forecast information held in long-term memory, but they can make precise predictions based on recent data. RNNs, as a result, do not provide enough performance as the gap length rises. Lengthy-term reliance of RNNs is addressed by LSTM networks, which can store information for a long period of time. LSTM networks are made up of a chain structure that includes four neural networks and several memory units known as cells. The cells store information, while the gates manage memory. The following sections explain the three main types of gates:

- Forget gate: This gate ignores any information that is irrelevant to the cell state. The gate receives the current input and preceding cell output, which are weighted before bias is applied. The result is sent into an activation function, which outputs a binary value. If the output for a particular cell state is zero, the information is lost; if the output is one, the information is saved for future use.
- Input gate: This gate adds useful information to the cell state. First, the information is regulated using the *sigmoid* function and values are filtered using the current input and the previous cell output. The *tanh* function is then used to generate a vector that contains all of the potential values, with output ranging from -1 to $+1$. Finally, the values of the vector are multiplied by the controlled values to produce the usable information.
- Output gate: This gate extracts useful information from the current state of the cell. First, a vector of values is generated by applying the *tanh* function on the cell. The *sigmoid* function is then used to control the data, and the values are filtered using the current input and prior cell output. Finally, the values are multiplied by the regulated values and sent as an output and input to the next cell.

2) biLSTM layer: Bidirectional LSTM (biLSTM) is the modified LSTM which has a bidirectional flow to process a sequence in both forward and backward direction and fed forward to the output layer [30]. Two hidden layers are present in biLSTM to compute hidden sequences both in forward and backward direction and to update the output layer by using backward layer (from last time step to the first) and forward layer (from first to last time step). This can improve the performance of LSTM networks by allowing future samples to provide context for past samples in a sequence of samples.

B. *Training the model*

A backpropagation approach followed by an optimization algorithm is used to estimate the weights and biases of the multi-neuron layers of the biLSTM model. The task of training the model is the same as that of minimizing a loss function, which is a measure of how well our biLSTM model performs in a classification test. The intuitive way to train the model is concluded in three steps, namely, (1) initialization of weights and biases, (2) evaluation of the model based on the estimated weights and biases, and the loss function, and (3) updating of the estimated weights and biases in the direction of finding a loss function minima. The loss function will be modest if its minima is as small as possible. In this case, our network performs very well. The backpropagation computes gradients, which are then employed by the training optimization algorithm to reduce the loss function. Despite the fact that there are numerous loss functions, they all fundamentally penalize us based on the distance between the predicted value from a given value and the actual value in our dataset. The mean squared error (MSE) is a widespread type of loss function. This error distance is easily estimated by taking all the errors, square their lengths, and find their average.

C. Training optimization algorithms

Stochastic gradient descent (SGD) selects a few samples randomly from the training features to estimate the network weights and biases. SGD only takes into account the first-order derivatives of the loss function, which means it has no clue about the curvature of the loss function. It can tell whether the loss is declining and how fast, but cannot differentiate between whether the curve is a plane, curving upwards or curving downwards. The solution is to consider the second-order derivative, or the rate of how quickly the gradient is changing. The Newton's method is a famous strategy that uses second-order derivatives to fix this issue. This is accomplished by computing the Hessian matrix, which is a matrix of the second-order derivatives of the loss function with respect to all combinations of the weights. The Hessian requires you to compute the gradients of the loss function with respect to every combination of weights. The number of parameters for modern day architectures may be in the billions, and calculating a billion squared gradients renders higher order optimization approaches computationally intractable.

- SGDM : Momentum is a prominent approach that is used in conjunction with SGD. Rather than relying solely on the gradient of the current step to guide the search, the momentum considers the gradient of previous steps to identify the best course of action. This enables us to get closer to the minima of the loss function faster. As a result, momentum is also referred to in our search as a mechanism for dampening oscillations. It increases speed and accelerates convergence, however, you should utilize simulated annealing if you overshoot the minima. In practice, the momentum coefficient is set at 0.5 and gradually annealed to 0.9 across training epochs.
- RMSprop: The RMSprop (root mean squared propagation) optimizer is comparable to the SGDM optimizer. RMSProp aims to attenuate oscillations similarly to momentum, but in a different method. RMSprop also eliminates the need to manually modify the learning rate by doing so automatically. In addition, RMSProp selects a distinct learning rate for each parameter. It is also worth noting that RMSProp performs simulated annealing by default. Assume we are approaching the minima of the loss function and want to slow down, so we do not overshoot the minima of the loss function. RMSProp automatically will decrease the size of the gradient steps towards the minima when the steps are too large.
- ADAM: So far, RMSProp and momentum have taken opposing approaches. While momentum speeds up our search for the minima of the loss function, RMSProp slows down our hunt for oscillations. The heuristics of both momentum and RMSProp are combined in ADAM, or adaptive moment optimization algorithms. It scales the learning rate using squared gradients, similar to RMSprop, and it takes advantage of momentum by using the moving average of the gradient rather than the gradient itself, similar to SGD with momentum.

Despite the fact that ADAM appears to be the most promising on paper, SGD with momentum may be the most common of the three training optimization techniques. Given the same loss function, empirical results indicate that all of these techniques can converge to various optimal local minima. SGD with momentum, on the other hand, seems to find flatter minima than ADAM, whereas adaptive approaches converge quickly to sharper minima. As a result, flatter minima generalise better than sharper ones, as it will be demonstrated experimentally in the following section.

III. Experimental Analysis

A. Dataset

The datasets that have been analyzed here in this article, as well as a few other research cited, are selected from the PhysioNet 2016 challenge freely accessible on the Web [27]. Heart sound recordings were recorded by placing an electronic stethoscope at four different locations on the chest. The challenge dataset contains heart sound recordings labeled as either normal or abnormal. We have selected a balanced dataset comprises 300 recordings for 150 healthy hearts and 150 unhealthy hearts. The length of the recordings varies from 5 to 120 seconds, with a sampling rate of 2000 Hz.

B. Preprocessing

Heart sound recordings obtained using diagnostic tools are usually contaminated with noise from various sources. These sounds hinder the early detection of mild heart sound recordings. Thus, filtering noise to remove such artifacts becomes an essential issue. This should be done at the cost of preserving all diagnostic information required for analysis of PCG signals, but removing all unwanted entities called noise. The PCG signals are heavily filtered to remove the noise from the sound.

In general, the heartbeat S_1 is the transient low-frequency signal, which is mainly between 10 and 200 Hz, produced by the vibrations of heart chambers, heart valves, and blood in the systolic. The heartbeat S_2 is produced at the end of systole, after the closure of the semilunar valves about the aortic and pulmonary. The beat S_2 has a higher pitch than the beat S_1, with its frequency range between 20 and 250 Hz. A 15^{th}-order Butterworth low-pass filter with a cut-off frequency of 250 Hz is used for this purpose. Filtering removes the high-frequency noise, and keeps low-frequency diagnostic information.

Furthermore, for the reduction of both calculations and computation complexity, the filtered signals are down-sampled by the factor 4, just before the extraction of classification features, following the Nyquist theorem.

C. Training and testing

The down-sampled PCG signals are truncated to 10 seconds (5000 samples) and a Gaussian sliding window of length 75 ms is applied to the truncated signals for short-time feature extraction. The entire features of the dataset are divided into 70% for training the model and 30% for testing it. The training and testing features are chosen at random, and the model is trained and tested independently thirty times. The classification performance is calculated by averaging the results of thirty experiments. Furthermore, the classification results are reported as a function of the optimization algorithms for four different values for the number of hidden neurons in the biLSTM model.

D. Model implementation

Table I lists values of the parameters for training the model. These parameters include general ones that make use in all other optimization algorithms, and specific ones that are assigned values in a certain algorithm. The initial learning rate, the learning rate drop factor, and the number of epochs are given values in all other optimization algorithms. Other parameters such as the momentum is assigned a value in SGDM, the squared gradient decay in RMSprop, and the gradient decay in ADAM.

E. Classification performance

The classification performance is probably of the highest interest in the evaluation of classification systems. Obviously, it should be a measure of how many signal examples were correctly classified and how many signal examples were incorrectly classified. Two possible errors can occur: a false negative (FN) indicating that the total number of PCG signals of abnormal hearts that are declared as normal, and a false positive (FP) which is the total number of PCG signals of normal hearts that are declared as abnormal. Similar to the definition of FP and FN, the true positives (TP) are the correctly identified abnormal heart sounds and the true negatives (TN) are the correctly classified normal heart sounds.

The classification performance of PCG signals is generally measured using the above quantities by computing the sensitivity (Sens.), the specificity (Spec.), and the accuracy (Acc.) as [31]:

$$Sensitivity = \frac{TP}{TP + FN} 100\%$$

$$Specificity = \frac{TN}{TN + FP} 100\%$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} 100\%$$

F. Results

The experimental classification performance is reported in detail in table II. The results show how different system settings affect performance. The experiments are carried out by varying the values of several parameters of the biLSTM classification model. In this way, the biLSTM model is compared to three distinct optimization algorithms, each with four different numbers of hidden neurons. Overall, the SGDM optimization technique is more effective than the other two for training the biLSTM. This is supported by the sensitivity, specificity, and accuracy scores for each algorithm, and this also matches the theory in which the flatter minima found by SGDM generalizes better than the sharper ones found by RMSprop and ADAM.

TABLE I
IMPLEMENTATION PARAMETERS OF THE BILSTM MODELS.

Parameters	SGDM	RMSprop	ADAM
Initial learning rate	0.01	0.001	0.001
Learning rate drop	0.90	0.90	0.90
L2-regularization	0.0001	0.0001	0.0001
Max epochs	500	500	500
Momentum	0.90	–	–
Squared gradient decay	–	0.90	–
Gradient decay	–	–	0.90

The biLSTM achieves the best classification accuracy of 89.10, which surpasses the baseline method provided in [13] (i.e., between 74.07 and 81.40). This value of accuracy is yielded by the SGDM training algorithm for a number of neurons less than or equal 50. When we train a high number of neurons, we get model overfitting, which causes performance loss. Furthermore, the ADAM algorithm performs better than the RMSprop optimizer for a small number of neurons of the biLSTM network, but not for larger numbers.

IV. CONCLUSIONS

Finding an optimum ANNs model that can effectively classify PCG signals while also optimizing numerous parameters requires an adequate training optimization algorithm that could handle such an optimization challenge. In this paper, we proposed an experimental study on the optimization algorithms for training the biLSTM network for the classification of PCG signals. We evaluated three different algorithms, namely, the stochastic gradient with momentum (SGDM) algorithm, the root mean squared propagation (RMSprop) algorithm, and the adaptive moment (ADAM) algorithm. Although ADAM appears to be the most promising algorithm in theory, SGDM is likely to be the most popular of the three. SGDM seems to find flatter minima than ADAM, but adaptive approaches converge quickly to sharper minima. As a result, smoother minima generalizes better than sharper ones. The elaborated results carried out in this study are in match with the theory. In this regard, the biLSTM classifier performs better when trained using the SGDM algorithm rather than the other two algorithms. These experiments were conducted using labeled heart sound recordings selected from the PhysioNet 2016 dataset. Each PCG signal of the selected recordings is represented by ten sequences of statistical features. Training sequences were used to train four different biLSTM network models, each with a different number of hidden neurons. The classification performance concluded that when a large number of neurons are trained, model overfitting occurs, resulting in performance reduction. Furthermore, for a small number of neurons in the biLSTM network, the ADAM method outperforms the RMSprop optimizer, but not for larger numbers.

TABLE II
THE AVERAGE RESULTS FOR THE DATASET OF THE PHYSIONET 2016 CHALLENGE. THE OVERALL ACCURACY OF THE BASELINE METHOD IS FROM 74.07 TO 81.40 [13].

# neurons	5			30			50			100			Average		
Opt.	Sens.	Spec.	Acc.	Sens.	Spec.	Acc.	Sens.	Spec.	Acc.	Sens.	Spec.	Acc.	Sens.	Spec.	Acc.
SGDM	89.30	88.90	89.10	92.90	85.20	89.10	92.90	85.20	89.10	85.70	81.50	83.60	90.20	85.20	87.73
RMSprop	64.30	70.40	67.30	64.30	70.40	67.30	82.10	85.20	83.60	78.60	74.10	76.40	72.33	75.03	73.65
ADAM	67.90	81.50	74.50	75.00	81.50	78.20	75.00	74.10	74.50	71.40	77.80	74.50	72.33	78.73	75.43

REFERENCES

[1] R. Rangayyan, R. Lehner, Phonocardiogram signal analysis: a review., Critical Reviews in Biomedical Engineering 15 (1987) 211–236.

[2] M. S. Obaidat, Phonocardiogram signal analysis: techniques and performance comparison., Journal of medical engineering technology 176 (1993) 221–7.

[3] F. Castells, P. Laguna, L. Sörnmo, A. Bollmann, J. Millet-Roig, Principal component analysis in ecg signal processing, EURASIP Journal on Advances in Signal Processing 2007 (2007) 1–21.

[4] L. M. Sepúlveda-Cano, E. Gil, P. Laguna, G. Castellanos-Domínguez, Selection of nonstationary dynamic features for obstructive sleep apnoea detection in children, EURASIP Journal on Advances in Signal Processing 2011 (2011) 1–10.

[5] S. Nemati, A. Malhotra, G. D. Clifford, Data fusion for improved respiration rate estimation, EURASIP Journal on Advances in Signal Processing 2010 (2010) 1–10.

[6] M. El-Segaier, O. Lilja, S. Lukkarinen, L. Sörnmo, R. Sepponen, E. Pesonen, Computer-based detection and analysis of heart sound and murmur, Annals of Biomedical Engineering 33 (2005) 937–942.

[7] S. M. Debbal, F. Bereksi-Reguig, Detection of differences of the phonocardiogram signals by using the continuous wavelet transform method, in: SOCO 2013, 2013.

[8] J. R. Deller, J. H. L. Hansen, J. G. Proakis, Discrete-Time Processing of Speech Signals, 2nd Edition, IEEE Press, New York, 2000.

[9] J. R. Bulgrin, B. J. Rubal, C. R. Thompson, J. M. Moody, Comparison of short-time fourier, wavelet and time-domain analyses of intracardiac sounds., Biomedical sciences instrumentation 29 (1993) 465–72.

[10] J. bo Wu, S. X. Zhou, Z. H. Wu, X.-M. Wu, Research on the method of characteristic extraction and classification of phonocardiogram, 2012 International Conference on Systems and Informatics (ICSAI2012) (2012) 1732–1735.

[11] G. Redlarski, D. Gradolewski, A. Palkowski, A system for heart sounds classification, in: PloS one, 2014.

[12] M. Hamidi, H. Ghassemian, M. Imani, Classification of heart sound signal using curve fitting and fractal dimension, Biomed. Signal Process. Control. 39 (2018) 351–359.

[13] S. A. Singh, S. Majumder, Classification of unsegmented heart sound recording using knn classifier, Journal of Mechanics in Medicine and Biology 19 (2019) 1950025.

[14] N. Andrisevic, K. Ejaz, F. Rios-Gutierrez, R. Alba-Flores, G. Nordehn, S. Burns, Detection of heart murmurs using wavelet analysis and artificial neural networks, Journal of Biomechanical Engineering 127 (6) (2005) 899.

[15] J. P. de Vos, M. M. Blanckenberg, Automated pediatric cardiac auscultation, IEEE Transactions on Biomedical Engineering 54 (2) (2007) 244–252.

[16] I. Grzegorczyk, M. Solinski, M. Lepek, A. Perka, J. Rosinski, J. Rymko, K. Stepien, J. Gieraltowski, PCG classification using a neural network approach, in: 2016 Computing in Cardiology Conference (CinC), Computing in Cardiology, 2016.

[17] H. Tang, H. Chen, T. Li, M. Zhong, Classification of normal/abnormal heart sound recordings based on multi-domain features and back propagation neural network, in: 2016 Computing in Cardiology Conference (CinC), Computing in Cardiology, 2016.

[18] M. Abdollahpur, A. Ghaffari, S. Ghiasi, M. J. Mollakazemi, Detection of pathological heart sounds, Physiological Measurement 38 (8) (2017) 1616–1630.

[19] M. Tschannen, T. Kramer, G. Marti, M. Heinzmann, T. Wiatowski, Heart sound classification using deep structured features, in: 2016 Computing in Cardiology Conference (CinC), Computing in Cardiology, 2016.

[20] V. Maknickas, A. Maknickas, Recognition of normal–abnormal phonocardiographic signals using deep convolutional neural networks and mel-frequency spectral coefficients, Physiological Measurement 38 (8) (2017) 1671–1684.

[21] J. Rubin, R. Abreu, A. Ganguli, S. Nelaturi, I. Matei, K. Sricharan, Recognizing abnormal heart sounds using deep learning, CoRR abs/1707.04642 (2017).

[22] S. Latif, M. U. Usman, J. Qadir, R. Rana, Abnormal heartbeat detection using recurrent neural networks, CoRR abs/1801.08322 (2018).

[23] L. C. Jain, L. R. Medsker, Recurrent Neural Networks: Design and Applications, CRC Press, Inc., USA, 1999.

[24] S. Alam, R. Banerjee, S. Bandyopadhyay, Murmur detection using parallel recurrent & convolutional neural networks, CoRR abs/1808.04411 (2018).

[25] S. Khaled, M. Fakhry, A. Mubarak, Classification of pcg signals using a nonlinear autoregressive network with exogenous inputs (narx), Proceedings of 2020 International Conference on Innovative Trends in Communication and Computer Engineering, ITCE 2020 (2020) 98–102.

[26] S. Sun, Z. Cao, H. Zhu, J. Zhao, A survey of optimization methods from a machine learning perspective, IEEE Transactions on Cybernetics 50 (2020) 3668–3681.

[27] C. Liu, D. J. Springer, Q. X. Li, B. Moody, R. Juan, F. J. Chorro, F. Castells, J. M. Roig, I. Silva, A. E. W. Johnson, Z. Syed, S. E. Schmidt, C. D. Papadaniil, L. Hadjileontiadis, H. Naseri, A. Moukadem, A. Dieterlen, C. Brandt, H. Tang, M. Samieinasab, M. R. Samieinasab, R. Sameni, R. G. Mark, G. D. Clifford, An open access database for the evaluation of heart sound algorithms., Physiological measurement 37 12 (2016) 2181–2213.

[28] A. Lerch, An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics, Wiley-IEEE Press, 2012.

[29] G. V. Houdt, C. J. Mosquera, G. Nápoles, A review on the long short-term memory model, Artificial Intelligence Review (2020) 1–27.

[30] M. Schuster, K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (1997) 2673–2681.

[31] J. Marôco, D. Silva, A. Rodrigues, M. Guerreiro, I. Santana, A. de Mendonça, Data mining methods in the prediction of dementia: A real-data comparison of the accuracy, sensitivity and specificity of linear discriminant analysis, logistic regression, neural networks, support vector machines, classification trees and random forests, BMC Research Notes 4 (2011) 299 – 299.