2019 6th International Conference on Instrumentation, Control, and Automation (ICA)
Bandung, Indonesia. 31 July – 2 August 2019

Deep Learning with Long Short-Term Memory for Enhancement Myocardial Infarction Classification

Annisa Darmawahyuni
Faculty of Computer Science,
University of Sriwijaya
Palembang, Indonesia
riset.annisadarmawahyuni@gmail.com

Siti Nurmaini*
Intelligent System Research Group
Faculty of Computer Science,
University of Sriwijaya
Palembang, Indonesia
siti_nurmaini@unsri.ac.id

Sukemi
Faculty of Computer Science,
University of Sriwijaya
Palembang, Indonesia
sukemi@ilkom.unsri.ac.id

***Abstract*—Myocardial infarction (MI) may be a minor event in a type of chronic disease, even undetectable. However, it can also be a major disaster that causes sudden death. The multivariance in ECG signals for different patients causes the interpretation of existence MI is a difficult task. The various conventional method is proposed to diagnose MI of ECG signals. The conventional classifier algorithm uses a shallow feature learning architecture based on the hand-crafted feature. This paper is only a preliminary study so that this paper contains only brief analysis and plan. However, it can present other point-of-view to process cardiac rhythm that associated in timesteps based on deep learning approach. Basically, a shallow feature learns as well as deep learning. However, the advantage and characteristics of deep learning will make classifier learn automatically without having to involve human intervention. Long short-term memory (LSTM) as deep learning classifier is proposed to the binary classification of MI and healthy control patients. The public ECG signals dataset of Physionet is used to support our research. In the evaluation of binary classification, balanced accuracy (BAcc) and Matthew's Correlation Coefficient (MCC) metrics are used to analyze imbalance sequential data of 4.57 Imbalance Ratio (IR). The overall, 3 hidden LSTM layers as classifier show good performance in imbalanced data to classify MI with precision, sensitivity, F1 score, BAcc, and MCC is 0.91, 0.91, 0.90, 0.83, and 0.75 respectively.**

Keywords—myocardial infarction, recurrent neural network, long short-term memory, ECG

I. INTRODUCTION

Cardiovascular disease (CVD) is the leading global cause of death in low- and middle- income countries [1]. It estimated to be the first-leading killer in 2020 [2]. The result of Basic Health Research at the Indonesian Ministry of Health in 2013 explained that the most common CVD in adults is coronary heart disease. The estimated prevalence of coronary heart disease in Indonesia is around 883.447 or 0.5 percent [3]. Among several types of coronary heart disease, myocardial infarction (MI) is the most dangerous form of coronary heart disease with the highest mortality rate [4]. MI occurs due to a lack of oxygen demand in the cardiac muscle tissue which requires the high supply of oxygen, which reduces the supply of oxygen to the area. To pump blood throughout the body, oxygen is needed by the cardiac muscles. If cardiac muscle activity increases, oxygen demand also increases [5].

MI can be diagnosed via an electrocardiogram (ECG) examination [5]. ECG is the cardiac electric activity recording via electrodes placed on the surface of the human body [6]. ECG signals have five different waveforms for each cardiac cycle; P wave, QRS complex, and T wave [7][8]. In normal patients, the five waveforms show the appropriate shape, no morphology [9][10]. However, in patients with MI, ECG changes can be observed typically of the ST interval length, ST elevation and changes in T waveform [11]. Interpretation of patients with MI with the ST interval length, ST elevation and changes in T waveform via ECG signals are a difficult task [12]. It is due to the significant variation of ECG signals morphological for different patients under different physical conditions [13].

The variety of conventional algorithms have been proposed to automatically interpret MI from ECG signal recordings [14]-[19]. These algorithms use a machine learning approach. The main drawback of machine learning results is the use of heuristic engineered features with shallow feature learning architectures [20]. In the steps to overcome the lack of it, this study gives other perspectives to analyze ECG signal that associated in timesteps based on deep learning approach. In recent years, a deep learning algorithm has shown a superior classification compared to conventional methods belong to a machine learning approach [21]. Deep learning learns features automatically and their own computational methods, without conventional hand-crafted features [20].

Some deep learning techniques used for ECG signal processing include convolutional neural networks [22], stacked autoencoders [23], deep belief network [24], deep Boltzmann machine [25], recurrent neural network [26][27], and long short-term memory [28]-[30] by entering different data characteristics, such as signal morphology, time-series, and sequential data in ECG. Among these algorithms, the recurrent neural network with long short-term memory (LSTM) architecture as a classifier works to process sequential data [28]-[30]. This sequential data in line with the nature of sequential ECG signals which assume that inputs and outputs depend on each other with or without a concrete notion of time [31]. The characteristic of the LSTM algorithm does not assume IID (independent and identically distributed) data.

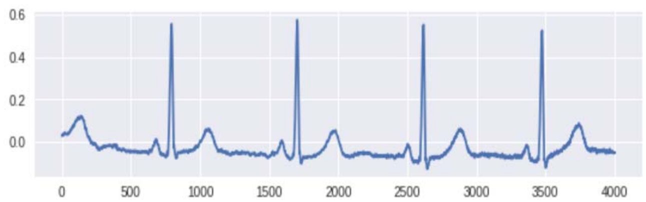
To produce a good analysis of classifier performance, this study uses balanced accuracy and Matthew's Correlation Coefficient due to the data ratio imbalanced; major (MI) and minor (healthy control) classes. These metrics are terminology and derivations from a confusion matrix in the diagnostic test.

This paper is organized as follows. Section 2 describes the material in our proposed method and classification problem. Section 3 explains the experimentation of some validation model in a classifier. Section 4 discusses the results of the classifier performance. The last, section 5 shows the conclusions of this work.

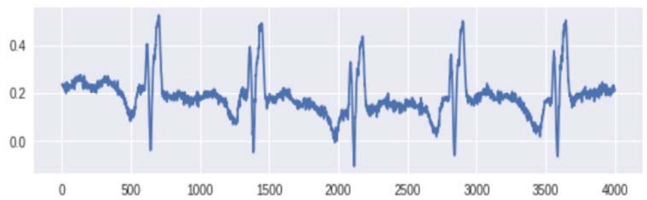
II. Material and Method

A. Data Preparation

The Physionet PTB Diagnostic ECG Database (PTBDB) contains 549 records from 290 patients [32]. Each ECG signal has been digitized at 1000 samples per second in PTBDB. In ECG signal pre-processing, the initial process of segmentation to achieve a fixed window size of 4 seconds. Each window includes at least four heartbeats at a normal heart rate, as shown in Fig. 1. The length of each kind of ECG signal is different. The range of the MI ECG signal length from 480000 to 1800180 samples (480 - 1800 seconds). For the length of the ECG signal in the normal heart with a range from 1455000 to 1800180 samples (1455 - 1800 seconds).



(a) Healthy Control



(b) Myocardial Infarction

Fig. 1. ECG Signal Morphological

B. Recurrent Neural Network

Recurrent Neural Network (RNN) has "memory" namely state (s_t) that captures information about all input elements (x_t) to output $\hat{y}_t$ [33]. Output vector h_t is used to compute error with loss function. In standard RNN, also known as vanilla RNN, has the same forward pass and backward pass process as other artificial neural networks. In the backpropagation process, the term being backpropagation through time (BPTT) [34]. The forward pass (black arrow) and backward pass (dotted line) can be illustrated in Fig. 2.

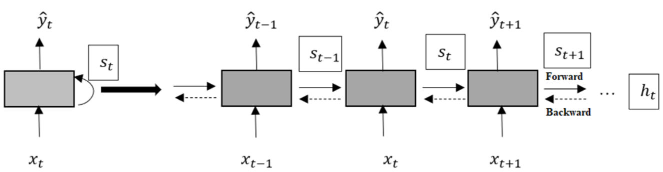


Fig. 2. Forward and Backward pass in standard RNN [35]

Singh et al., proposed RNN technique for Arrhythmia Classification [35]. Theoretically, a standard RNN can handle input long-term dependencies, but in practice, learning issue appears due to vanishing or exploding gradients problems in backward pass [36]. In backward pass, more derivation is often calculated, resulting in smaller number and even close to the value 0 (zero). It often disappear when it reaches the initial layers. A total error in all timesteps T, in weight W the error is given by the (1):

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W} \tag{1}$$

The overall error gradient in (1) can be calculated by the chain rules is given by (2):

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E}{\partial y_t} \frac{\partial y_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial W} \tag{2}$$

Differentiation of h_t and h_k in (2) is a derivative of a hidden state that stores memory at time t, which is related to the hidden state at the previous time k. This phase involves the Jacobians $\frac{\partial h_i}{\partial h_{i-1}}$ for the entire time t and one-time k:

$$\frac{\partial h_t}{\partial h_k} = \frac{\partial h_t}{\partial h_{t-1}} \frac{\partial h_{t-1}}{\partial h_{t-2}} \dots \frac{\partial h_{k+1}}{\partial h_k} = \prod_{i=k+1}^{t} \frac{\partial h_i}{\partial h_{i-1}} \tag{3}$$

The Jacobian matrix $\frac{\partial h_t}{\partial h_{t-1}}$ displays the eigen decomposition given by $W^T diag[f'(h_{t-1})]$, the eigenvalues are produced $\lambda_1, \lambda_2, \dots, \lambda_n$ where $|\lambda_1| > |\lambda_2| \dots |\lambda_n|$ and corresponds to eigenvectors $v_1, v_2, \dots, v_n$. If the largest eigenvalue produced is larger than 1, $\lambda_i < 1$, there will be vanishing gradient, on the contrary, if the largest value is smaller than 1, $\lambda_i > 1$, then there will be an exploding gradient. To overcome vanishing and exploding gradient problems on standard RNN, this study proposes Long Short-Term Memory (LSTM) architecture.

C. Long Short-Term Memory Model

Long Short-Term Memory (LSTM) is a specific type of RNN architecture, which is used to solve vanishing or exploding gradient problems [37][38]. LSTM architecture improves the performance of a standard RNN model by entering a gate mechanism for handling timestep information from input sequences at long intervals. Using this gating mechanism, LSTM overcomes vanishing or exploding gradients, wherein there is no gate in a standard RNN [33]. The gate mechanism controls the amount of information, from the previous timestep, which contributes to the current output. The LSTM gate mechanism implements three components; (1) inputs, (2) forget, and (3) output gate [33].

Forward Pass

A forward pass is calculated as input x with a length T by starting $t = 1$ and recursively by applying an update equation while adding t. The scripts i, f and o refer to the input, forget, and output gates from the block, respectively. Script c refers to one of the C memory cells. At time t, LSTM receives a new input in the form of vector x^t (including bias), and the output of the vector h^{t-1} in the previous timesteps (see Fig. 3, which $\otimes$ denotes elementwise product Hadamard).

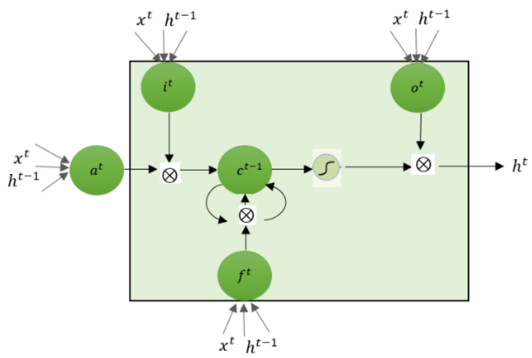


Fig. 3. Forward pass in LSTM

Weights from cell c to input, forget, and output gates are annotated w_i, w_f, w_o, respectively. The equations are given by:

$$a^t = tanh(W_c x^t + U_c h^{t-1}) \tag{4}$$

$$i^t = \sigma(W_i x^t + U_i h^{t-1}) = \sigma(\hat{\imath}^t) \tag{5}$$

$$f^t = \sigma\left(W_f x^t + U_f h^{t-1}\right) = \sigma(\hat{f}^t) \tag{6}$$

$$o^t = \sigma(W_o x^t + U_o h^{t-1}) = \sigma(\hat{o}^t) \tag{7}$$

Ignoring the non-linearities:

$$z^t = \begin{bmatrix} \hat{a}^t \\ \hat{\imath}^t \\ \hat{f}^t \\ \hat{o}^t \end{bmatrix} = \begin{bmatrix} W^c & U^c \\ W^i & U^i \\ W^f & U^f \\ W^o & U^o \end{bmatrix} \cdot \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} = W.I^t \tag{8}$$

Then, the memory cell values updated by combining a^t and the contents of the previous cell c^{t-1}. The combination is based on the magnitude of the gate input i^t dan forget gate f^t:

$$c^t = i^t \odot a^t + f^t \odot c^{t-1} \tag{9}$$

In the end, the LSTM cell calculates the output value by passing an updated cell value through non-linearity:

$$h^t = o^t \odot f(c^t) \tag{10}$$

Backward Pass

Backward pass computes starting from $t = T$, and recursively calculating the derivative unit in each timestep. As standard RNN, all status and activations are initialized to zero at $t = 0$, and all $\delta = 0$ at $t = T + 1$. The output of the forward pass in (10), in differentiation, is given by:

$$\delta h^t = \frac{\partial E}{\partial h^t} \tag{11}$$

Then, applying the chain rule:

$$\frac{\partial E}{\partial o_i^t} = \frac{\partial E}{\partial h_i^t} \cdot \frac{\partial h_i^t}{\partial o_i^t} = \delta h_i^t \cdot f(c_i^t) \tag{12}$$

where,

$$\delta o^t = \delta h^t \odot f(c^t) \tag{13}$$

An update memory cell in the backward pass process based on the forward pass in (9), in differentiation, is given by:

$$\delta c^t = \frac{\partial E}{\partial c^t} \tag{14}$$

$$\frac{\partial E}{\partial i_i^t} = \frac{\partial E}{\partial c_i^t} \cdot \frac{\partial c_i^t}{\partial i_i^t} = \delta c_i^t \cdot a_i^t \tag{15}$$

where,

$$\delta i^t = \delta c^t \odot a^t \tag{16}$$

Computation of input and gate in the backward pass phase based on forward pass in (8), then to get δz^t the parameter $\delta\hat{a}^t, \delta\hat{\imath}^t, \delta\hat{f}^t$, and $\delta\hat{o}^t$ are needed as following:

$$\delta\hat{a}^t = \delta a^t \odot (1 - tanh^2(\hat{a}^t)) \tag{17}$$

$$\delta\hat{\imath}^t = \delta i^t \odot i^t \odot (1 - i^t) \tag{18}$$

$$\delta\hat{f}^t = \delta f^t \odot f^t \odot (1 - f^t) \tag{19}$$

$$\delta\hat{o}^t = \delta o^t \odot o^t \odot (1 - o^t) \tag{20}$$

Then, produces:

$$\delta z^t = [\delta\hat{a}^t, \delta\hat{\imath}^t, \delta\hat{f}^t, \delta\hat{o}^t]^T \tag{21}$$

In calculating the forward pass in (8) where $z^t = W.I^t$, using the parameter δz^t, then $\delta I^t = W^t \cdot \delta z^t$, so:

$$I^t = \begin{bmatrix} x^t \\ h^{t-1} \end{bmatrix} \tag{22}$$

where δh^{t-1} can be taken from δI^t and produces:

$$\delta W^t = \delta z^t \cdot (I^t)^T \tag{23}$$

Differentiation in the backward pass from (8) to (23) can be represented in Fig. 4.

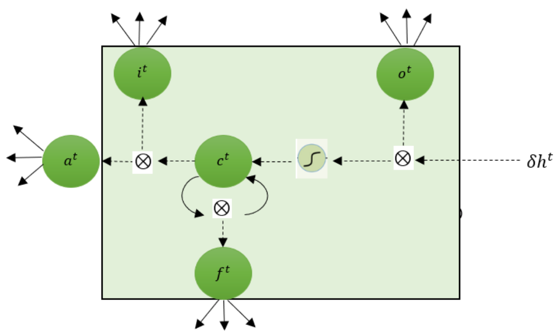


Fig. 4. Backward pass in LSTM

III. EXPERIMENTAL

The amount of ECG signal data is divided by 80% for the training set and 20% for the testing set. In the training set, 10% is used for the validation set. Total of 12.359 ECG rhythm that has been segmented each window sized in 4 seconds. The amount of total data is separated randomly. The number of sequential data for the class of MI is 10.144 and the healthy control class is 2.215 of the total.

The stages of the learning phase in neural networks are validation. Evaluation parameters used in the validation process in the binary classification process between the class of MI and healthy control are using confusion matrix, which contains information about the actual classification and predictions made by the classification system. The validation phase starts from standard RNN, one to three hidden LSTM layers model.

In this study, hyper-parameters are used to optimize the performance of the LSTM architecture. Adam optimization with 0.001 learning rate, each classifier was trained for 50 epochs, and a batch size of 512 samples. All classifiers were trained on NVIDIA GeForce RTX 2080.

IV. EVALUATION PERFORMANCE

This study evaluates binary classification with balanced accuracy (BAcc) and Matthew's Correlation Coefficient (MCC) for classification in imbalanced data. The equations are given by:

$$BAcc = \frac{(\frac{TP}{P}+\frac{TN}{N})}{2} \tag{24}$$

$$MCC = \frac{(TP\ x\ TN)-(FP\ x\ FN)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{25}$$

The data in the classification process are divided into two different classes, namely positive (P) as Myocardial Infarction and negative (N) as Healthy control. This classification produces four types of results; two types of classifications that are true (or true), namely true positive (TP) and true negative (TN); and two types of false (or false) classifications, namely false positive (FP) and false negative (FN). The 2x2 table formulated with these four results is called a confusion matrix [10]. The basic evaluation steps of confusion matrix produce the following equations:

$$Sensitivity\ (Recall) = \frac{TP}{(TP+FN)} \tag{26}$$

$$Precision = \frac{TP}{(TP+FP)} \tag{27}$$

$$F1\ Score = \frac{2*Precision*Recall}{(Precision+Recall)} \tag{28}$$

V. RESULT AND ANALYSIS

In this section, we discuss the four models (standard RNN, 1 hidden LSTM, 2 hidden LSTM, and 3 hidden LSTM layers) results that have been tested on the testing set. The results are represented in Table I.

TABLE I. PERFORMANCE RESULTS OF BINARY CLASSIFICATION IN TESTING SET

Model	Class	Performance		
		Precision	*Sensitivity*	*F1 Score*
Standard RNN	Healthy Control	0.00	0.00	0.00
	MI	0.83	1.00	0.91
	avg/total	**0.68**	**0.83**	**0.75**
1 hidden LSTM layer	Healthy Control	0.91	0.59	0.71
	MI	0.91	0.99	0.94
	avg/total	**0.91**	**0.91**	**0.90**
2 hidden LSTM layers	Healthy Control	0.85	0.65	0.73
	MI	0.92	0.97	0.94
	avg/total	**0.90**	**0.91**	**0.90**
3 hidden LSTM layers	Healthy Control	0.94	0.68	0.74
	MI	0.90	0.99	0.94
	avg/total	**0.91**	**0.91**	**0.90**

Table I shows that the results of the classification performance in healthy control class are lower than the MI class in each model. It is not strange considering the amount of healthy control data available is less than the MI of the total data. It can result in an increase in the number of false positives (FP) and a decrease in the number of false negatives (FN). Table II shows the performance results of binary classification with the imbalance ratio (IR) equals 4.57. The IR is defined as the size ratio of the large class and the small class.

TABLE II. PERFORMANCE RESULT OF BINARY CLASSIFICATION WITH IR = 4.57 IN TESTING SET

Model	Performance	
	BAcc	*MCC*
Standard RNN	0.50	0.00
1 hidden LSTM layer	0.78	0.68
2 hidden LSTM layers	0.81	0.68
3 hidden LSTM layers	0.83	0.75

VI. CONCLUSION

This paper is an initial stage to determine the structure and algorithm of LSTM as a classifier of ECG signals. This does not take away the fact that the results of the performance in this initial stage have not been too good due to minimum ECG signal pre-processing before being classified by LSTM. In addition, the LSTM architecture used is still standard and simple. Even though it is not too optimal, for a simple LSTM network, the results of performance in the training and testing sets show good classification when compared to RNN standard. This proves that LSTM as deep learning technique is a pretty good method for classifying sequential data that implements timesteps in imbalanced data.

ACKNOWLEDGMENT

This work is supported by the Kemenristek Dikti Indonesia under Basic Research Fund Number. 096/SP2H/LT/DRPM/2019, and Universitas Sriwijaya, Indonesia under Hibah Unggulan Profesi Fund 2019.

REFERENCES

[1] C. J. L. Murray, A. D. Lopez, W. H. Organization, and others, “Global comparative assessments in the health sector: disease burden, expenditures and intervention packages,” 1994.

[2] A. Nursalim, M. Suryaatmadja, and M. Panggabean, “Potential clinical application of novel cardiac biomarkers for acute myocardial infarction,” *Acta Med Indones*, vol. 45, no. 3, pp. 240–250, 2013.

[3] B. Penelitian, "Riset kesehatan dasar," *Jakarta Kementeri. Kesehat. RI*, 2013.
[4] I. Alwi, "Infark Miokard Akut Dengan Elevasi ST," *Buku Ajar Ilmu Penyakit Dalam*, pp. 1741–1756, 2009.
[5] K. Thygesen, J. S. Alpert, H. D. White, and others, "Universal definition of myocardial infarction," *J. Am. Coll. Cardiol.*, vol. 50, no. 22, pp. 2173–2195, 2007.
[6] R. A. Ricardo, R. A. Bassani, and J. W. M. Bassani, "A simple laboratory method for teaching how electrocardiogram is generated," in *World Congress on Medical Physics and Biomedical Engineering, September 7-12, 2009, Munich, Germany*, 2009, pp. 385–387.
[7] J. S. Fleming, *Interpreting the electrocardiogram*. Springer Science & Business Media, 2012.
[8] Siti Nurmaini, Radiyati Umi Partan, Muhammad Naufal Rachmatullah, "Deep Neural Networks Classifiers on The Electrocardiogram Signal for Intelligent Interpretation System," *Sriwij. Int. Conf. Med. Sci.*, 2018.
[9] S. Nurmaini, A. Gani, and others, "Cardiac Arrhythmias Classification Using Deep Neural Networks and Principle Component Analysis Algorithm.," *Int. J. Adv. Soft Comput. Its Appl.*, vol. 10, no. 2, 2018.
[10] A. Darmawahyuni, "Coronary Heart Disease Interpretation Based on Deep Neural Network," *Comput. Eng. Appl. J.*, vol. 8, no. 1, 2019.
[11] P. J. Zimetbaum and M. E. Josephson, "Use of the electrocardiogram in acute myocardial infarction," *N. Engl. J. Med.*, vol. 348, no. 10, pp. 933–940, 2003.
[12] S. Mawri *et al.*, "The comparison of physician to computer interpreted electrocardiograms on ST-elevation myocardial infarction door-to-balloon times," *Crit. Pathw. Cardiol.*, vol. 15, no. 1, pp. 22–25, 2016.
[13] S. Banerjee and M. Mitra, "Application of cross wavelet transform for ECG pattern analysis and classification," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 326–333, 2014.
[14] A. Keshtkar, H. Seyedarabi, P. Sheikhzadeh, and S. H. Rasta, "Discriminant analysis between myocardial infarction patients and healthy subjects using Wavelet Transformed signal averaged electrocardiogram and probabilistic neural network," *J. Med. Signals Sens.*, vol. 3, no. 4, p. 225, 2013.
[15] P. Kora and S. R. Kalva, "Improved Bat algorithm for the detection of myocardial infarction," *Springerplus*, vol. 4, no. 1, p. 666, 2015.
[16] N. A. Bhaskar, "Performance analysis of support vector machine and neural networks in detection of myocardial infarction," *Procedia Comput. Sci.*, vol. 46, pp. 20–30, 2015.
[17] R. S. Remya, K. P. Indiradevi, and K. K. A. Babu, "Classification of myocardial infarction using multi resolution wavelet analysis of ECG," *Procedia Technol.*, vol. 24, pp. 949–956, 2016.
[18] M. Kumar, R. B. Pachori, and U. R. Acharya, "Automated diagnosis of myocardial infarction ECG signals using sample entropy in flexible analytic wavelet transform framework," *Entropy*, vol. 19, no. 9, p. 488, 2017.
[19] L. D. Sharma and R. K. Sunkaria, "Inferior myocardial infarction detection using stationary wavelet transform and machine learning approach," *Signal, Image Video Process.*, vol. 12, no. 2, pp. 199–206, 2018.
[20] B. Pyakillya, N. Kazachenko, and N. Mikhailovsky, "Deep learning for ECG classification," in *Journal of physics: conference series*, 2017, vol. 913, no. 1, p. 12004.
[21] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.
[22] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," *arXiv Prepr. arXiv1707.01836*, 2017.
[23] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096–1103.
[24] M. Huanhuan and Z. Yue, "Classification of electrocardiogram signals with deep belief networks," in *Computational Science and Engineering (CSE), 2014 IEEE 17th International Conference on*, 2014, pp. 7–12.
[25] S. M. Mathews, C. Kambhamettu, and K. E. Barner, "A novel application of deep learning for single-lead ECG classification," *Comput. Biol. Med.*, vol. 99, pp. 53–62, 2018.
[26] E. D. Übeyli, "Combining recurrent neural networks with eigenvector methods for classification of ECG beats," *Digit. Signal Process.*, vol. 19, no. 2, pp. 320–329, 2009.
[27] E. D. Übeyli, "Recurrent neural networks employing Lyapunov exponents for analysis of ECG signals," *Expert Syst. Appl.*, vol. 37, no. 2, pp. 1192–1199, 2010.
[28] H. W. Lui and K. L. Chow, "Multiclass classification of myocardial infarction with convolutional and recurrent neural networks for portable ECG devices," *Informatics Med. Unlocked*, vol. 13, pp. 26–33, 2018.
[29] N. Strodthoff and C. Strodthoff, "Detecting and interpreting myocardial infarctions using fully convolutional neural networks," *arXiv Prepr. arXiv1806.07385*, 2018.
[30] S. Goto *et al.*, "Artificial intelligence to predict needs for urgent revascularization from 12-leads electrocardiography in emergency patients," *PLoS One*, vol. 14, no. 1, p. e0210103, 2019.
[31] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.
[32] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215--e220, 2000.
[33] O. Faust, A. Shenfield, M. Kareem, T. R. San, H. Fujita, and U. R. Acharya, "Automated detection of atrial fibrillation using long short-term memory network with RR interval signals," *Comput. Biol. Med.*, vol. 102, pp. 327–335, 2018.
[34] J. A. Bullinaria, "Recurrent neural networks," 2015.
[35] S. Singh, S. K. Pandey, U. Pawar, and R. R. Janghel, "Classification of ECG Arrhythmia using Recurrent Neural Networks," *Procedia Comput. Sci.*, vol. 132, pp. 1290–1297, 2018.
[36] D. Rav`i *et al.*, "Deep learning for health informatics," *IEEE J. Biomed. Heal. informatics*, vol. 21, no. 1, pp. 4–21, 2017.
[37] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[38] A. Darmawahyuni, S. Nurmaini, W. Caesarendra, V. Bhayyu, M. N. Rachmatullah, and others, "Deep Learning with a Recurrent Network Structure in the Sequence Modeling of Imbalanced Data for ECG-Rhythm Classifier," *Algorithms*, vol. 12, no. 6, p. 118, 2019.