August 2021

Dissertation for Ph.D.

ECG Recognition Using Deep Recurrent Neural Networks For Personal Identification

Graduate School of Chosun University

Department of Information and Communication Engineering

Beom-Hun Kim

ECG Recognition Using Deep Recurrent Neural Networks For Personal Identification

개인 식별을 위한
심층 순환 신경망 기반의 ECG 인식

August 27, 2021

Graduate School of Chosun University

Department of Information and Communication Engineering

Beom-Hun Kim

ECG Recognition Using Deep Recurrent Neural Networks For Personal Identification

Advisor: Prof. Jae-Young Pyun

This thesis is submitted to the Graduate School of Chosun University in partial fulfillment of the requirements for the Doctor's degree in engineering.

April 2021

Graduate School of Chosun University

Department of Information and Communication Engineering

Beom-Hun Kim

This is to certify that the Doctor's thesis of

Beom-Hun Kim

has been approved by the examining committee for the thesis requirement for the Doctor's degree in engineering

Committee Chairperson **Prof. Suk-Seung Hwang**	황 석 승	(Sign)
Committee Member **Prof. Goo-Rak Kwon**	권 구 락	(Sign)
Committee Member **Dr. Eui-Sung Kang**	강 의 성	(Sign)
Committee Member **Prof. KyungBaek Kim**	김 경 백	(Sign)
Committee Member **Prof. Jae-Young Pyun**	변 재 영	(Sign)

June 2021

Graduate School of Chosun University

Table of Contents

List of Tables

List of Figures

Abstract

ECG Recognition Using Deep Recurrent Neural Networks For Personal Identification

Beom-Hun Kim
Advisor: Prof. Jae-Young Pyun, Ph.D.
Depart. of Info. and Comm. Eng.,
Graduate School of Chosun University

Securing personal authentication is a significant study in the security field. Specifically, there are two methods as fingerprinting and face recognition for personal authentication. But, these systems have some issues like fingerprinting forgery or environmental obstacles. To solve the spoofing or forgery identification trouble, various approaches have been taken into account involving iris, electrocardiogram (ECG), and hybrid methods. Recently, ECG identification has been widely studied in the field because of its unique character. For the personal identification using ECG recognition, there are conventional several methods such as linear discriminant analysis (LDA), principal component analysis (PCA), support vector machine (SVM), deep neural network (DNN), and recurrent neural network (RNN).

According to the particular studies, the best performance in ECG identification is the RNN method as against the other methods. However, the these methods

need a lengthy input signal for high classification accuracy. Therefore, these methods may not be used in a real-time system.

This thesis shows the bidirectional long short-term memory (LSTM)-based deep recurrent neural networks (DRNN) with the late-fusion technique for the real-time ECG-based biometrics classification.

A preprocessing procedure for quick classification of ECG like noise removal (or reduction), derivative filter, moving average filter, and normalization are preproposed. The public dataset: MIT-BIH Normal Sinus Rhythm (NSRDB) and MIT-BIH Arrhythmia (MITDB) are used for the experimental evaluation in the proposed method.

The proposed method shows that in NSRDB, the overall precision 100%, accuracy 100%, recall 100%, and F1-score 1. In the case of MITDB, the result shows the overall precision 99.8%, accuracy 99.8%, recall 99.8%, and F1-score 0.99. This experiments prove that the proposed method accomplishes an overall higher classification accuracy and efficiency against the conventional LSTM approach.

요 약

개인 식별을 위한
심층 순환 신경망 기반의 ECG 인식

김 범 훈

지도교수: 변 재 영

조선대학교 대학원 정보통신공학과

개인 인증 보안은 보안 분야에서 중요한 연구입니다. 특히 개인 인증에는 지문과 얼굴 인식이 사용되고 있습니다. 그러나 이러한 시스템은 지문 위조 또는 환경 장애와 같은 특정 문제로 어려움을 겪습니다. 위조 또는 스푸핑 식별 문제를 해결하기 위해 홍채, 심전도 (ECG), 및 하이브리드 방법과 관련된 다양한 접근 방식이 고려되었습니다.

최근 ECG 식별은 고유한 특성 때문에 널리 연구되고 있습니다. ECG 인식을 이용한 개인 식별을 위해 선형 판별 분석 (LDA), 주성분 분석 (PCA), 서포트 벡터 머신 (SVM), 심층 신경망 (DNN) 및 순환 신경망 (RNN)과 같은 기존의 여러 방법이 있습니다. 특정 연구에 따르면 RNN 모델은 다른 모델에 비해 ECG 식별에서 최고의 성능을 제공합니다. 그러나 이러한 방법은 높은 정확도를 위해 긴 입력 신호가 필요합니다. 따라서 이러한 방법은 실시간 시스템에 적용되지 않을 수 있습니다.

본 논문에서는 심전도 기반 생체 인식 식별 및 분류를 위한 실시간 시스템을

개발하기 위해 후기 융합을 통한 양방향 장단기 기억 셀 (LSTM) 기반 DRNN (deep recurrent neural network)을 사용하는 것을 제안합니다. 잡음 제거(또는 감소), 미분 필터, 이동 평균 필터, 정규화와 같은 ECG의 빠른 식별을 위한 전처리 절차를 제안합니다. MIT-BIH NSRDB (Normal Sinus Rhythm)와 MITDB (MIT-BIH Arrhythmia)의 두 가지 공개 데이터 세트를 사용하여 제안된 방법을 실험적으로 평가했습니다.

제안된 LSTM 기반 DRNN 모델에 따르면 NSRDB에서 전체 정밀도는 100%, 정확도는 100%, 재현율은 100%, F1 점수는 1이었습니다. MITDB의 경우 전체 정밀도는 99.8 %, 정확도는 99.8%, 재현율은 99.8%, F1 점수는 0.99입니다. 실험은 제안된 모델이 기존 LSTM 접근 방식과 비교하면 전반적으로 더 높은 분류 정확도와 효율성을 달성한다는 것을 보여줍니다.

Ⅰ Introduction

1.1 Research background

In recent years, varied studies including various basic methods have been carried out in biometric systems, like a fingerprint, iris, face, voice, and electrocardiogram (ECG). However, fingerprint and face recognition systems have some disadvantages in the safety of personal authentication. Disadvantages are environmental obstacles, or forgery, such as hair, glass, or light. Presently, voice recognition systems take commonly advantage of carrying out simple tasks, like changing the TV channel, turning the lights off or on, or making a phone call. But, voice recognition systems are not enough elaborate to be taken into account as a dependable solution in the authentication system because of the danger of spoofing with a recorded voice in place of the lawful voice. Therefore, to consider forgery or spoofing identification problems, other approaches have to be addressed, ECG, as suggested in this thesis. ECG is to record after measuring the electrical signal of the heartbeat of humans. For ECG biometric systems, using linear discriminant analysis (LDA), principal component analysis (PCA), support vector machine (SVM), deep neural networks (DNN), and different analysis methods have been widely researched and used to medical disease diagnosis and personal identification systems [1-4]. The foregoing methods well-known as existing ECG identification procedures are needed the feature extraction to get high accuracy in the classification procedures. The latest deep learning methods do not use conventional feature extraction. In addition, to gain high accuracy, the deep learning methods need lengthy input data. The conventional personal authentication system exploiting ECG can be shown as shown

in Figure 1.

Figure 1 presents the conventional personal authentication system exploiting ECG with a deep learning method. First of all, an ECG database of individuals is needed that includes all kinds of ECG signals. The ECG of the state of an individual means calmness, eating, sleeping, running, walking, and etc.

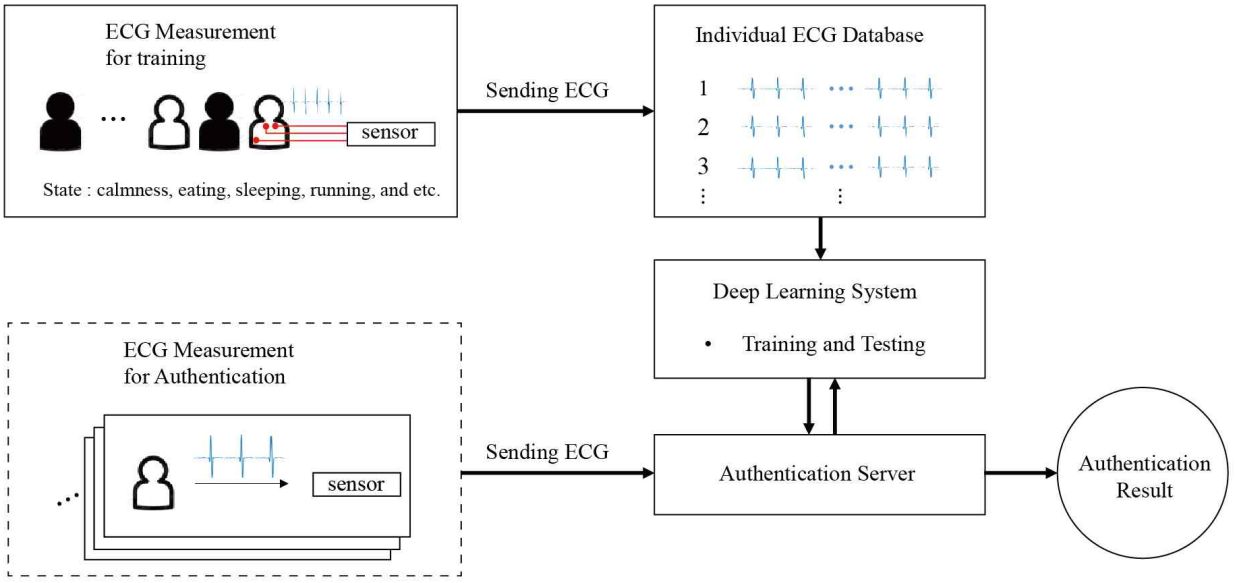


Figure 1. Conventional personal authentication system using ECG

And then, a deep learning system is trained and tested using the individual ECG database; accordingly, an authentication server completed. For personal authentication, the ECG input data from the dashed box in Figure 1, which is not trained and tested in the training and testing procedure, is sent to the server of authentication. The input ECG data are classified by the deep learning system into a certain user; consequently, the authentication is successful. This system for authentication may be applied for different self-certification services, like bank vaults, vehicles, and automated door locks.

1.2 Research objective

This thesis suggests the bidirectional long short-term memory (LSTM)-based deep recurrent neural networks (DRNN) to construct a system of ECG recognition for classifying the human ECG. The two public datasets from the Physionet database are used for the experimental evaluation using performance metrics in the proposed method [5]. The principal contributions of this thesis are as follows:

- The pre-processing procedures involving grouping the ECG signal of the short length, segmentation using a fixed segmentation time period, segmentation using R-peak detection, and non-feature extraction are proved. These procedures are taken into account authentication time for applying the real-time system.
- The bidirectional DRNN for ECG classification joined with the late-fusion technique is presented and performed. To the best of my belief, the proposed bidirectional DRNN model for personal classification has not been explained in the conventional literature.

1.3 Thesis organization

This thesis consist of as follows. Related analysis in the literature is reviewed in Chapter Ⅱ. The proposed pre-processing and proposed LSTM-based DRNN for ECG are explained in Chapter Ⅲ. Experimental results and concluding remarks are presented in Chapter Ⅳ and Ⅴ, respectively.

Ⅱ Background and Related work

2.1 Electrocardiogram (ECG)

2.1.1 ECG Principle of occurrence

The human heart is the central muscle organ of the circulatory system that circulate blood. It acts as a pump that supplies blood through the blood vessels to the entire body by repeated contraction and relaxation periodically. As shown in Figure 2, the heart consists of two atria (right atrium, left atrium), two ventricles (right ventricle, left ventricle), the sinoatrial node, atrioventricular node, bundle of His, bundle branch, Purkinje fibers, and cardiac muscle (atrial muscle; ventricular muscle). The atrium is responsible for accepting blood (the left atrium is from the lungs; the right atrium is from the upper and lower aorta). The right ventricle sends blood coming from the right atrium to the lungs through the pulmonary artery. The sinoatrial node can control the heartbeat according to the condition of the entire body, which means periodically transmitting electrical signals to regulate the contraction cycle of the heart. The sinoatrial node is situated on the upper wall of the right atrium. The heart voltage, which is created the fastest than other cells in the heart begins at the sinoatrial node.

The heart voltage spreads to both atria, contracting and relaxing the heart. It then reaches the atrioventricular node and takes approximately 0.1 seconds to pass through the atrioventricular node, which is delivered to the bundle of His. The electric current delivered to the bundle of His flows back to each branch, and the left and right branches are transmitted to the Purkinje fibers that are spread like a net throughout the left and right ventricular walls. The electric current spreads

through the Purkinje fibers to the ventricular muscles. The electricity generated at this time is referred to as an electrocardiogram [6-9].

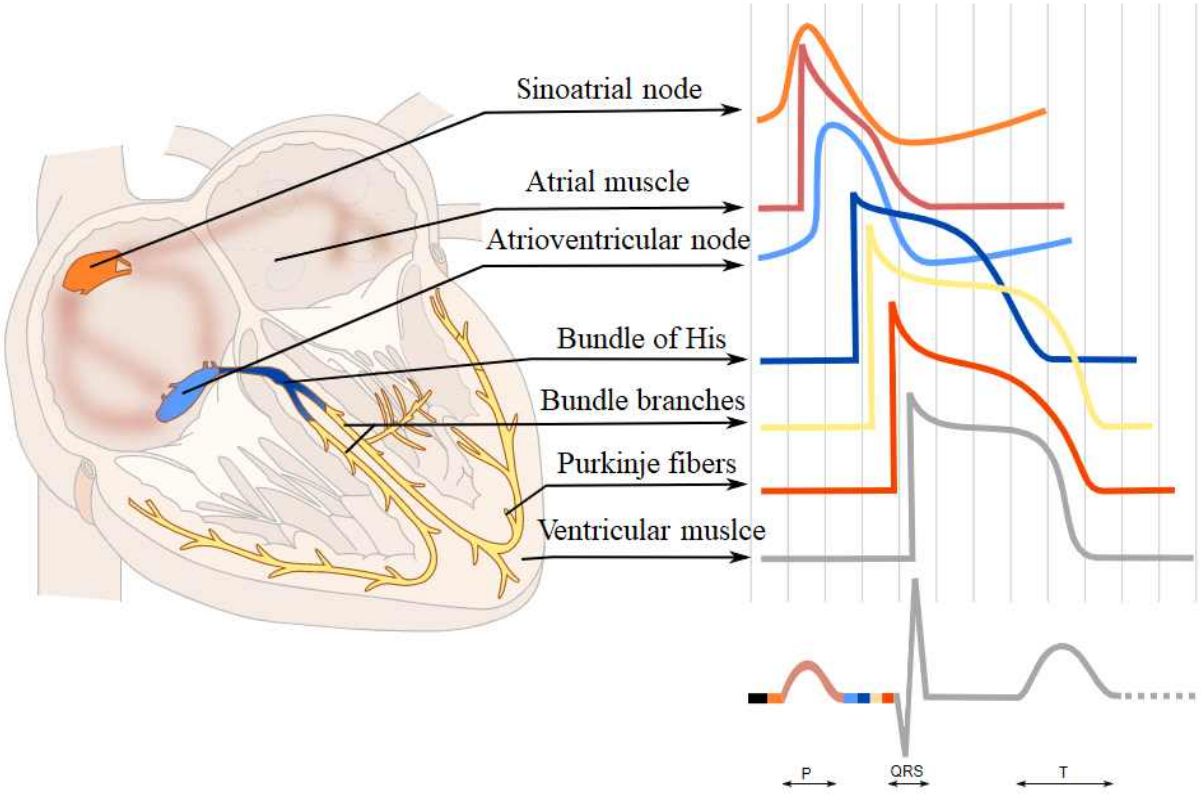


Figure 2. The structure of the heart and the construction of ECG cycle [10].

2.1.2 ECG signal

The measurement of the electrical signals of the heart was first attempted in 1903 by Willem Einthoven, a Dutch physiologist; the term electrocardiogram (ECG) was first used then. ECG refers to a weak current generated when the heart beats to generate a potential distribution on the body surface;, this potential change is measured and recorded on the body surface [11, 12]. Since then, ECG has been used as a significant element in the diagnosis of heart disease, and in recent years, research has been actively conducted as a signal for personal authentication.

The current generated in the sinoatrial node contracts the atrium, passes through the atrioventricular nodule, reaches the Purkinje fibers of the ventricle, and causes ventricular contraction. When the ventricle is filled with blood and the next current arrives, the ventricle contracts again. The depolarization (systolic phase) and repolarization (recovery phase) are referred to as cardiac cycles and take about 0.8 seconds.

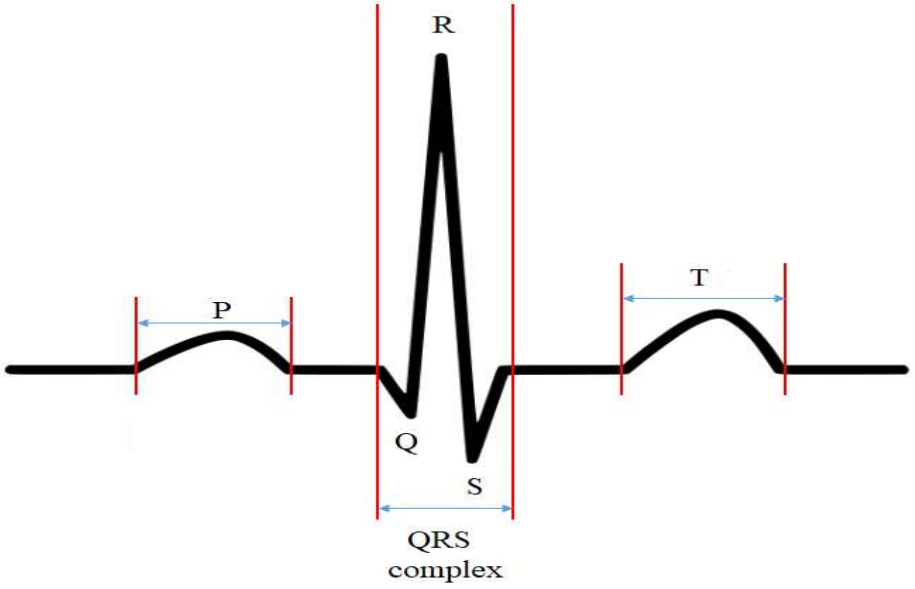


Figure 3. The waveform of the ECG

The electrical activity of the ECG cardiac cycle is expressed as a waveform, and the waveform of the ECG is composed of P, QRS (same as QRS complex), and T waves as shown in Figure 3.

The P wave is the atrial depolarization that occurs when electrical stimulation from the sinoatrial node contracts the atrium and has a gentle and round shape. The P wave takes 0.05 to 0.12 seconds and signals with amplitudes as small as 0.05 mV to 0.25 mV are measured. Particularly, the P-wave is associated with arrhythmia.

The P-R interval is from the starting point of the P wave to the starting point of the QRS complex and represents the time it takes for the electrical stimulation of the atrium to pass through the atrioventricular node to the bundle of His. The normal P-R interval is within 0.2 seconds. If the normal PR interval is delayed by more than 0.2 seconds, the conduction disturbance of the atrioventricular nodule is shorter than 0.12 seconds, which means that the contractile stimulation of the heart started in a place other than the sinoatrial node.

The QRS complex represents a depolarization state in which the ventricle contracts, and as a reference signal when measuring the heart rate, has a narrow and large amplitude. The normal period is 0.06 to 0.12 seconds, and signals with amplitudes of 1.5 mV are measured.

The Q-T interval is between the start point of the Q wave and endpoint of the T wave and represents the time from the start of ventricular depolarization to the end of repolarization. In the normal case, it takes 0.32 to 0.4 seconds.

The T wave is the repolarization of the ventricle, which indicates the recovery phase after ventricular contraction and has an asymmetrical shape. It is a signal with an amplitude of 0.5mV or less, takes 0.1 to 0.25 seconds, and when

abnormal, it becomes flat or inverted. There are individual differences because the T wave that occurs in the heart muscle. Table 1 presents a summary of the ECG characteristics. Additionally, a detailed configuration of the ECG of two cycles is shown in Figure 4.

Table 1. A summary of the ECG characteristic

Type	Definition	Duration (second)	Amplitude (mV)	Relation
P wave	Atrial depolarization	0.05~0.12	0.05~0.25	Arrhythmia
QRS complex	Ventricular depolarization	0.06~0.12	0.1~0.15	The standard of heart rate
T wave	Ventricular repolarization	0.1~0.25	0~0.5	Individual variation

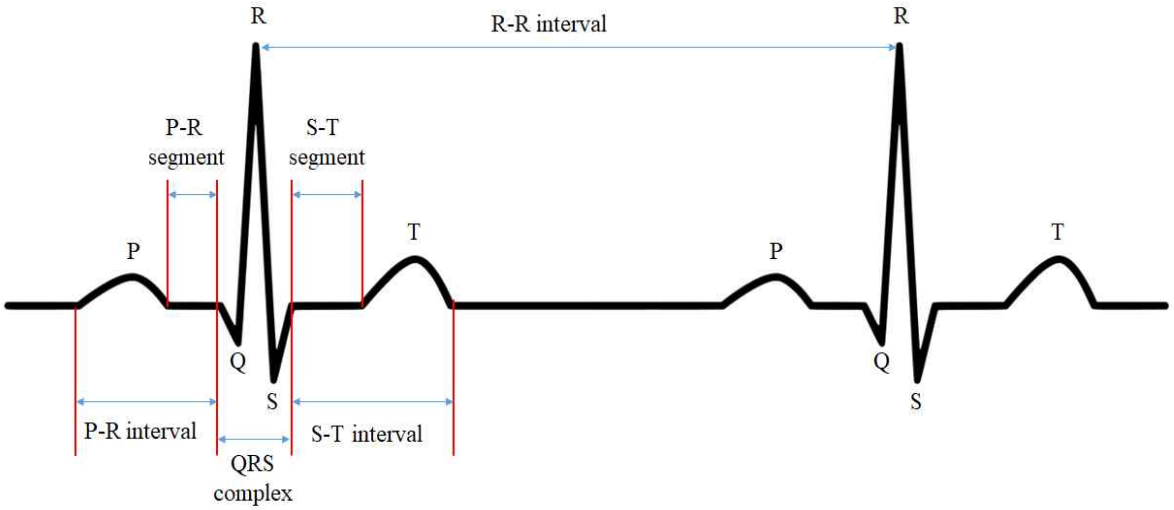


Figure 4. The detailed configuration from ECG of two cycles

2.2 Measurement method

2.2.1 Standard limb leads

The standard limb leads is the most basic electrocardiogram recording method, which is a method of attaching electrodes to two specific body parts and recording the voltage difference between the electrodes using an electrocardiogram. As show in Figure 5, LEAD Ⅰ is the potential difference between the left hand (+) and right hand (-). LEAD Ⅱ is the potential difference between the left foot (+) and right hand (-). When LEAD Ⅲ is the potential difference between the left foot (+) and left hand (-), LEAD Ⅱ can be expressed as the vector sum of LEAD Ⅰ and LEAD Ⅲ according to Einthoven's equilateral triangle law. The standard limb induction method can measure the electrical signal of the longitudinal section of the heart.

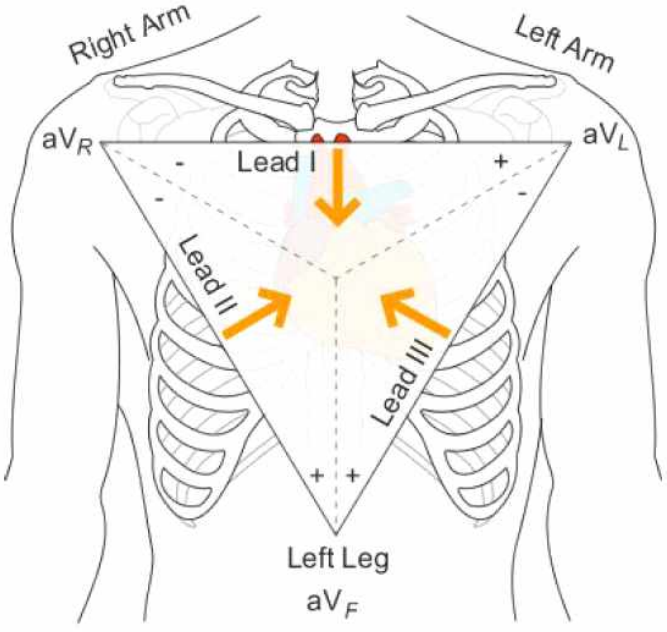


Figure 5. The ECG measurement principle by limb leads method [13].

2.2.2 Precordial lead

Chest leads represent the potential difference between the triangular electrical center electrode and the electrode mounted on the chest wall. As shown in Figure 6, six electrodes are attached to the chest wall as if surrounding the left side of the heart. V1 and V2 are the points where the 4th intercostal and right margin of the sternum, and 4th intercostal and left margin of the sternum meet, respectively. V1 and V2 are referred to as septal leads, and when the left arm is set to 0°, components of 120° and 90° are detected, respectively. V3 is the midpoint of V2 and V4, and V4 is the point where the 5th intercostal line and mid clavicle line meet; it is referred to as the anterior lead and detects components of 75° and 60°, respectively. V5 is located on the anterior axillary line at the same height as V4, and V6 is attached to the mid axillary line at the same height as V4. V5 and V6 are referred to as lateral leads, measuring the directions of 30° and 0°, respectively, as well as the cross-section of the heart.

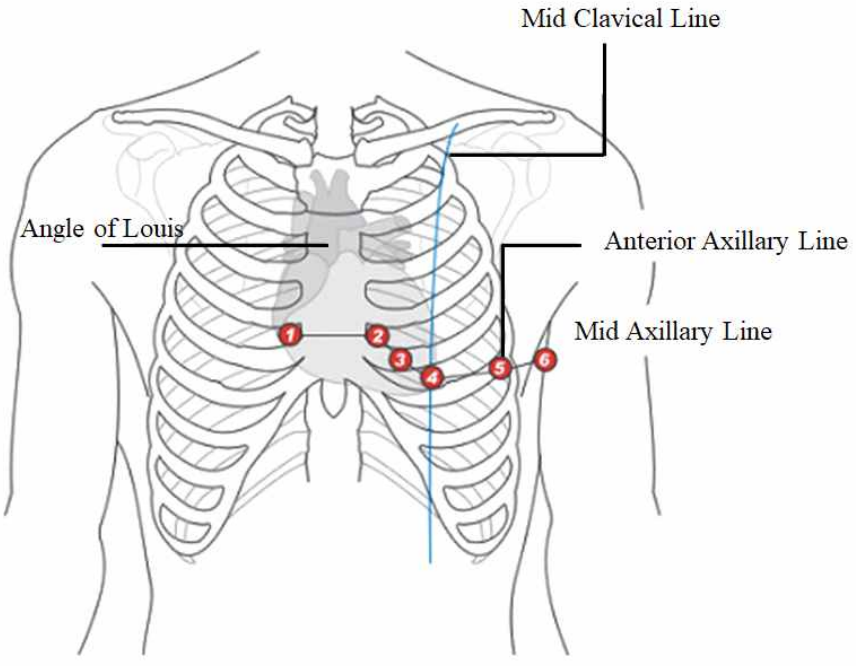


Figure 6. Electrode Location of precordial leads [13]

2.3 Basic preprocessing

2.3.1 Low and High pass filter

Generally, the ECG signal is distorted because of the breathing and power supply devices for ECG measurement. The frequency band of the ECG is mostly formed in the low band from 0.05 Hz to 100 Hz, thus, it is significantly affected by the low-frequency noise caused by respiration and movement. Therefore, to remove low and high-frequency noise, low and high pass filters are utilized. Even when low and high pass filters are employed, the noise of the ECG signal cannot be completely removed. Thus, additional signal processing methods suitable for this purpose are required.

In the case of a bandpass filter, it is used to detect the QRS complex of the ECG. Typically, a band of 2 ~ 58 Hz is set as the passband. The size of the passband is variably applied and used according to its purpose of use.

2.3.2 Moving Average

Moving average is a useful method for analyzing trends across data by creating a series of averages for several subsets of the entire data set. If a moving average is applied to time series data with many data fluctuation, it can find out the approximate trend of fluctuations in the time series data can be determined. This is one of the simplest and most widely used FIR filters in signal processing. The present output is created using the average of the past, present, and future values of the data. The shape of the output signal is influenced by the size of the subset to be averaged, and the degree of smoothness is determined. Therefore, it is

possible to obtain the effect of reducing the noise of all the data by newly composing the data while sorting by the average of a given data subset.

The moving average equation can be expressed as (1) and (2).

$$\bar{x}_k = \frac{x_{k-n+1} + x_{k-n+2} + \ldots + x_k}{n} \tag{1}$$

$$\bar{x}_{k-1} = \frac{x_{k-n} + x_{k-n+1} + \ldots + x_{k-1}}{n} \tag{2}$$

$\bar{x}_k$ indicates the average of a total of n data from $k-n+1$ th data to k th data. Based on (1) and (2), a moving average filter that calculates continuous values can be expressed as (3).

$$\bar{x}_k = \bar{x}_{k-1} + \frac{x_k - x_{k-n}}{n} \tag{3}$$

2.4 Classifier

2.4.1 Principal Component Analysis (PCA)

The principal component analysis (PCA) method refers to a method of reducing the dimension by taking the entire data and projecting it linearly along the axes of several eigen directions with large variances [14]. Reducing the dimension of the input vector through PCA transformation has the effect of data compression by maintaining information on the overall distribution of the data and removing noise parts that do not affect the variance. Furthermore, PCA is designed to reduce dimensions efficiently while retaining the original characteristics of data, thus, it can be regarded as an optimal linear transformation in terms of the mean square error (MSE). Additionally, when it is difficult to interpret the correlation between various data, it is possible to extract and analyze several independent principal components by linearly transforming the correlated variables.

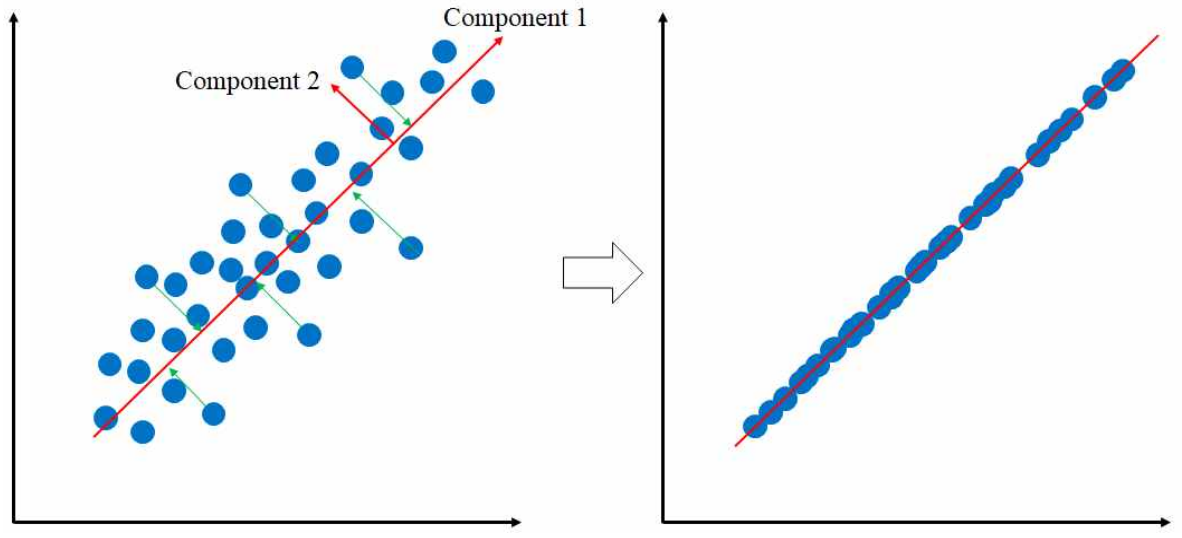


Figure 7. An example of principal component analysis

As shown in Figure 7, the direction with large variances, such as component 1 is found, and component 2 that is orthogonal to component 1 is found. Component 1 is the direction that contains the most information in the given data. In other words, the correlation of traits is the greatest direction. Component 2 is the direction with the most information among the directions perpendicular to component 1. The direction found through this process is referred to as the principal component because it is the direction of the main variance in the data. Consequently, the principal component analysis creates a new axis so that the variance represented by the data sample is maximized.

2.4.2 Linear Discriminant Analysis (LDA)

Linear discriminant analysis (LDA) is an algorithm that analyzes the decision hyperplane minimizing the variance of data within the class and maximizing the variance between classes. LDA has the advantage of relatively accurately classifying feature vectors between classes by maximizing class separation [15, 16]. In other words, the LDA method maximizes the deviation outside the group and minimizes the deviation within the group so that the data can be easily divided, and the group can also be easily separated. Additionally, because the location of the input data does not change, the line is drawn so that the given class can be clearly distinguished, thus it can prevent overlap between groups, as shown in Figure 8.

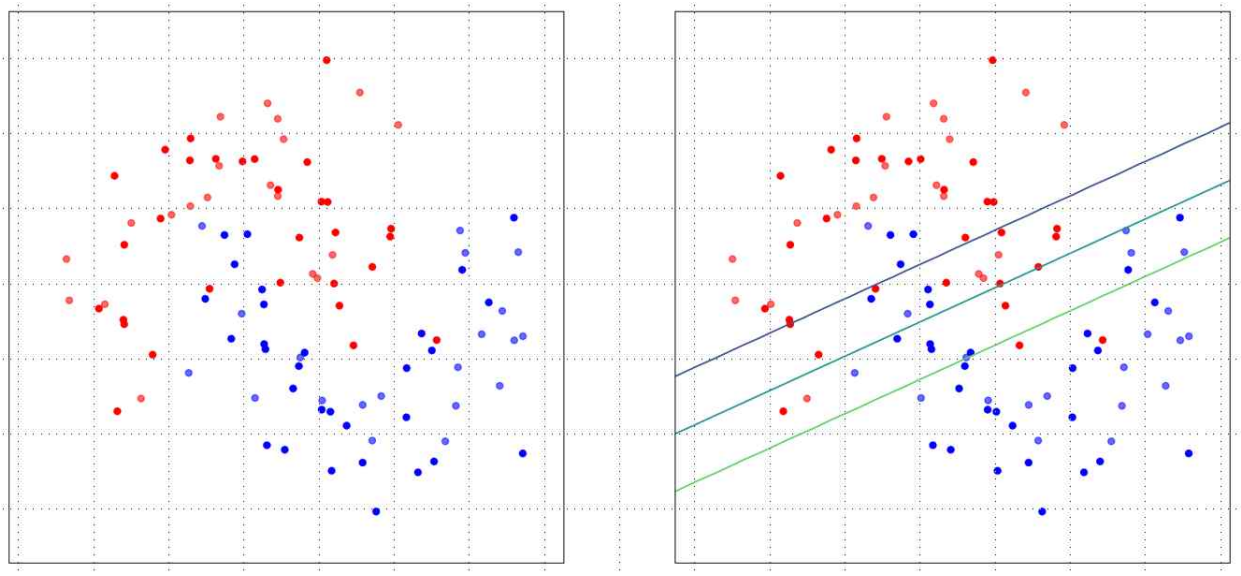


Figure 8. An example of linear discriminant analysis

2.4.3 Support Vector Machine (SVM)

The support vector machine (SVM) is an algorithm that creates a hyperplane with the maximum margin for each class to be classified from given data and classifies it based on the hyperplane when new data is given. The margin refers to the distance from the data closest to the decision boundary among the training data to the decision boundary. The data located closest to the decision boundary is the support vector. If the training data is fixed, the margin and support vector depend on the decision boundary. To reduce the generalization errors, maximizing the spacing between the two areas is recommended, so the decision boundary with the maximum margin is found. In the SVM, the linear crystal boundary with the maximum margin is referred to as a hyperplane, as shown in Figure 9. In this way, finding the maximum margin at the current dimension is referred to as linear SVM.

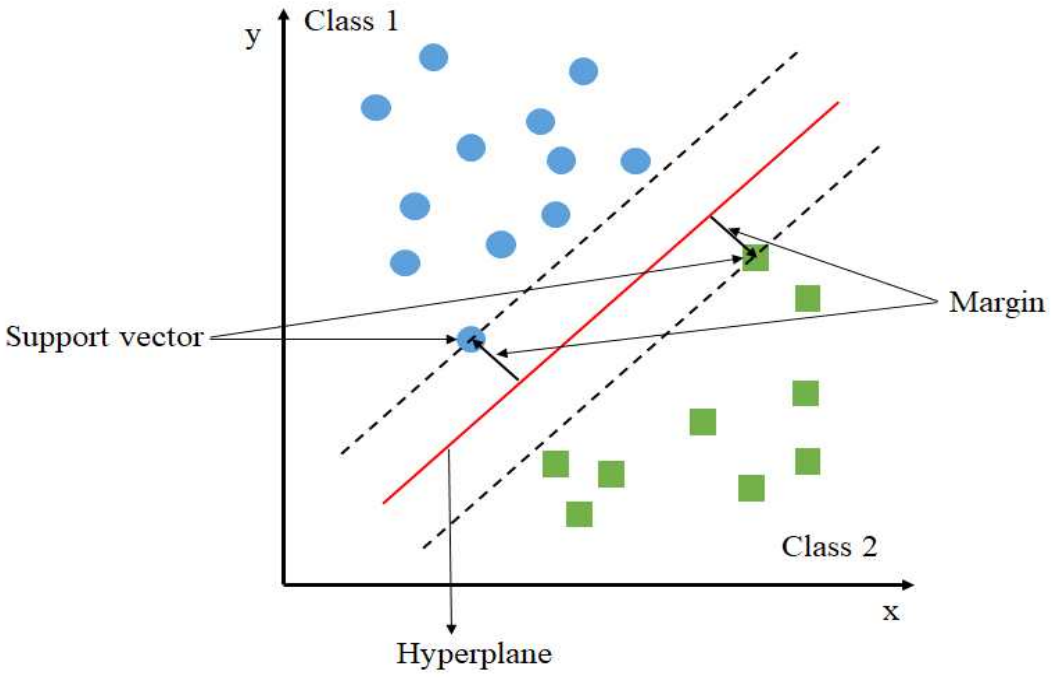


Figure 9. An example of Linear support vector machine classification

If the given data cannot be separated from the current dimension, it can be transformed to a higher dimension to create a fine hyperplane, as shown in Figure 10. This is referred to as nonlinear SVM. Nonlinear SVM increases the amount of computation in dimensional transformation. To solve this problem, a function known as kernel trick can be used to significantly reduce the amount of computation and apply it.

SVM has the advantage of classifying data probabilistically and most efficiently; however, if the estimation result is wrong, the next closest class cannot be known.

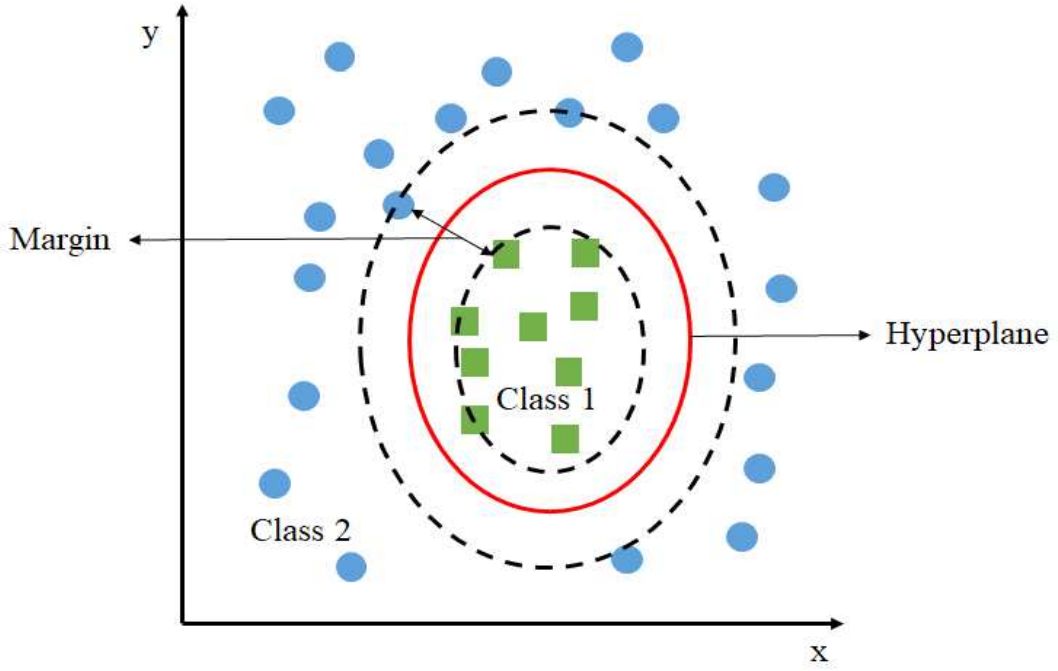


Figure 10. An example of nonlinear support vector machine classification

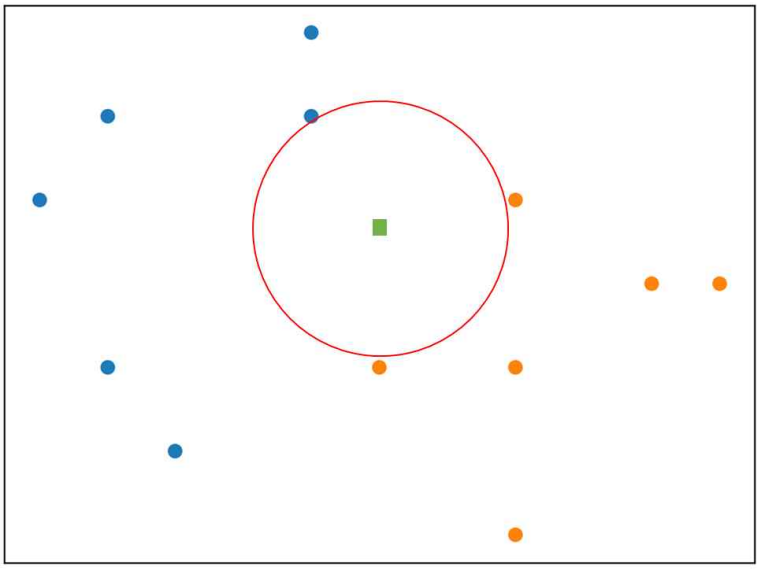
2.4.4 K-Nearest Neighbor (KNN)

The k-nearest neighbor (KNN) is an algorithm proposed by Cover, Hart in 1968 [17]. It is appended because it uses the k-nearest neighbors of the selected sample from a given sample set. It is an intuitive method to classify unlabeled samples by the distance between the data according to the similarity between the samples in the training data set. In other words, given a sample without a label, it finds the closest k-label samples in the training data set, and assigns the given sample to a group with a large number of frequencies to the set that appears when k is grouped. In the case of KNN's similarity measurement method, the result varies depending on the distance measurement method used based on the data flow and distribution. Table 2 lists examples of distance measurement methods. In most cases, the Euclidean distance is used to measure the degree of similarity. Since KNN is a nonparametric method, it can be used regardless of the distribution state of the sample.

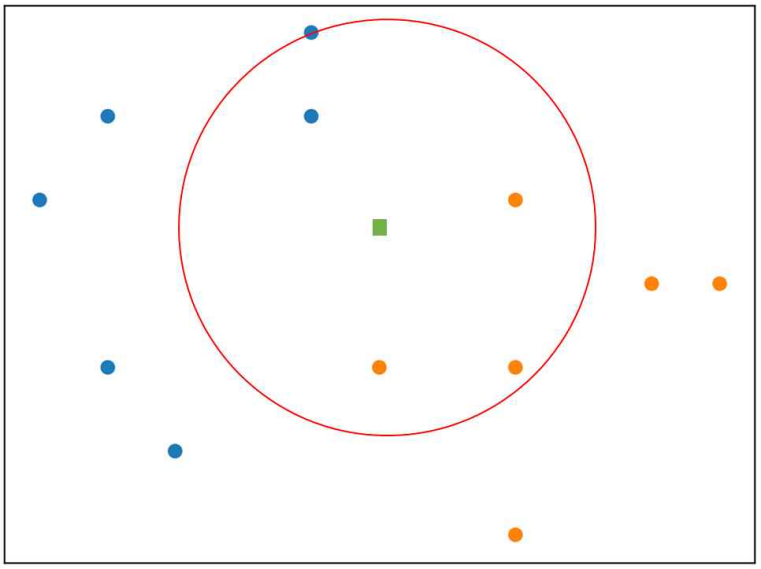
Table 2. Distance function

Distance function	Equation
Euclidean Distance	$D(X,Y)=\sqrt{\sum_{i}^{n}(x_i-y_i)^2}$
Manhattan Distance	$D=\sum_{i}^{n}\lvert x_i-y_i\rvert$
Minkowski Distance	$D=(\sum_{i}^{n}\lvert x_i-y_i\rvert^p)^{1/p}$
Mahalanobis distance	$d_M(x,y)=\sqrt{(x-y)^T\sum^{-1}(x-y)}$

If the value of K is small, it is classified finely and the probability of occurrence of a classification error increases. Conversely, if the value of K is large, the overall flow can be determined, but subdivided classification may not be possible. Therefore, the K value must be appropriately selected according to the number of data and number of classes to be classified, as shown in Figure 11.



(a) KNN classifier when K =1



(b) KNN classifier when K =5

Figure 11. KNN classification Examples

2.4.5 Neural Network (NN)

Neural networks (NN) are made by imitating neurons in human brain structures shown in Figure 12. The human brain is connected by a number of neurons and receives external signals (visual, auditory, and tactile), which are sent out as output (cognitive and behavioral) signals. Therefore, a neural network is a model in which neurons are connected to form a network, and an optimal output value corresponding to the input value is determined through mathematical operation. Each neuron constituting a neural network plays a role independently, and a single neuron does not significantly affect the results, which are derived through the combination of several neurons [18].

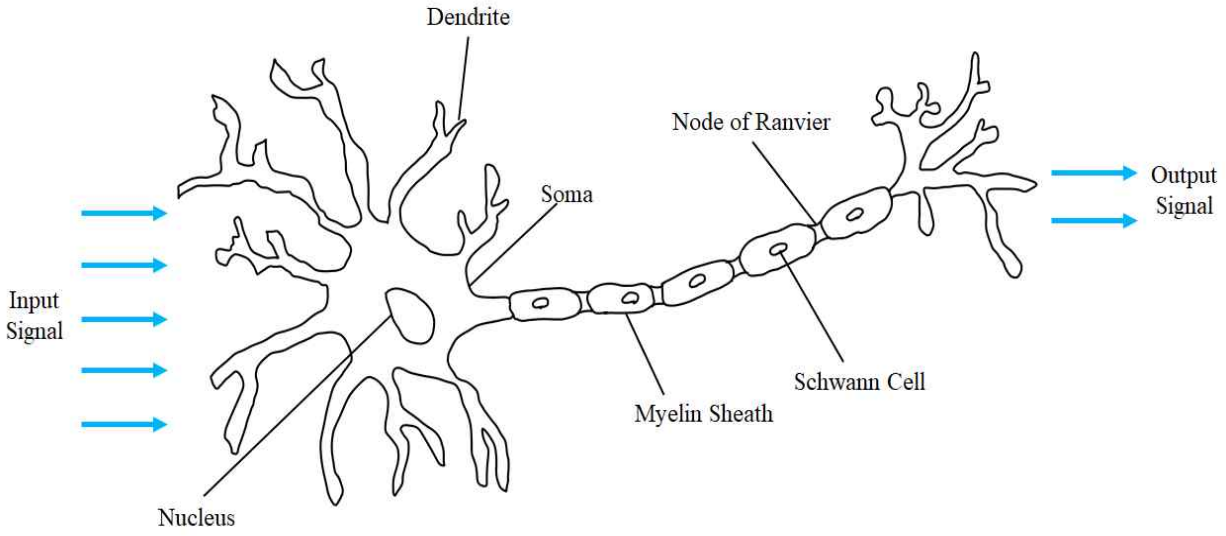


Figure 12. Understanding the brain structure of the human neural network

In a neural network, as shown in Figure 13, one neuron is calculated after multiplying each weight value when several input signals come in. It then creates its own output value through the following activation function.

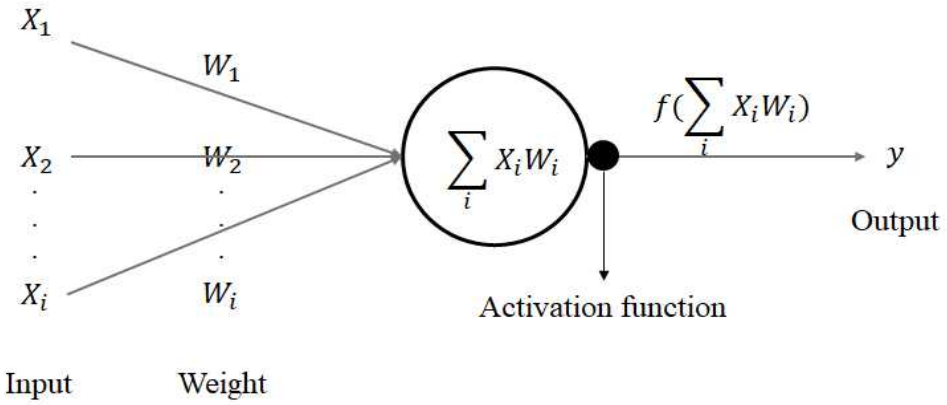


Figure 13. An example of a linear neuron

Neural networks can be classified by the data type, number of layers, output type, and activation function. It can be represented as a single-layer perceptron involving input and output layers as shown in Figure 13, and a multi-layer perceptron including input, hidden, and output layers as shown in Figure 14.

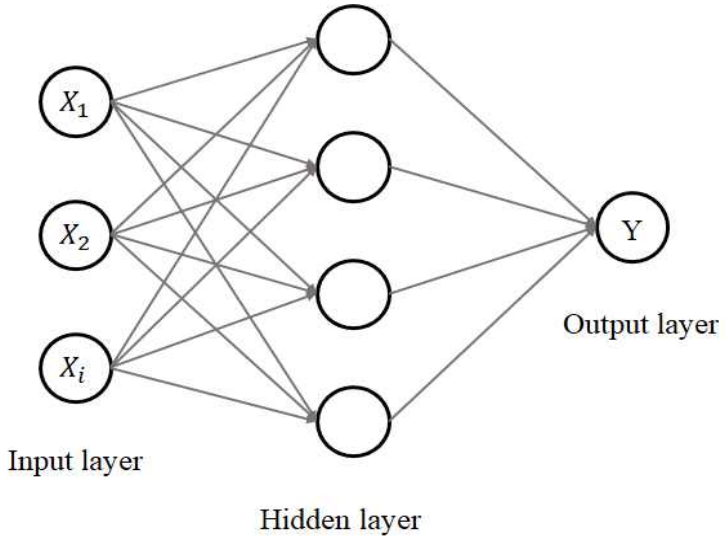


Figure 14. The structure of a multi-layer neural network

The activation function converts an incoming input value into a new output value. Representative activation functions are sigmoid and Rectified Linear Unit (ReLU). Sigmoid can be expressed by (4) and ReLU can be expressed as (5).

$$f(z) = \frac{1}{1+e^{-z}} \tag{4}$$

$$f(z) = \max(0, z) \tag{5}$$

z indicates the sum of the product of the input value and the weight as shown in Figure 13.

The output value of sigmoid has a value between 0 and 1, as show in Figure 15. The disadvantage of sigmoid is that the slope value converges to zero towards both ends. Therefore, the vanishing gradient phenomenon occurs in which the weights cannot be updated because the gradient decreases as it passes through several layers and cannot be propagated to the next layer.

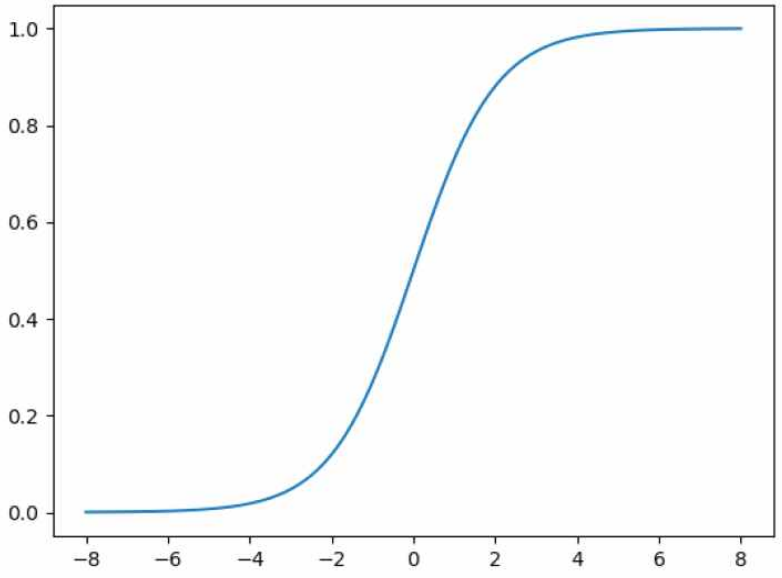


Figure 15. Sigmoid function according to z

ReLU is used to solve the vanishing gradient. ReLU has a constant slope of 1 when z is positive, solving the shortcomings of sigmoid as shown in Figure 16. Additionally, there is an advantage that the computational complexity is low, and the speed of convergence to learning is fast.

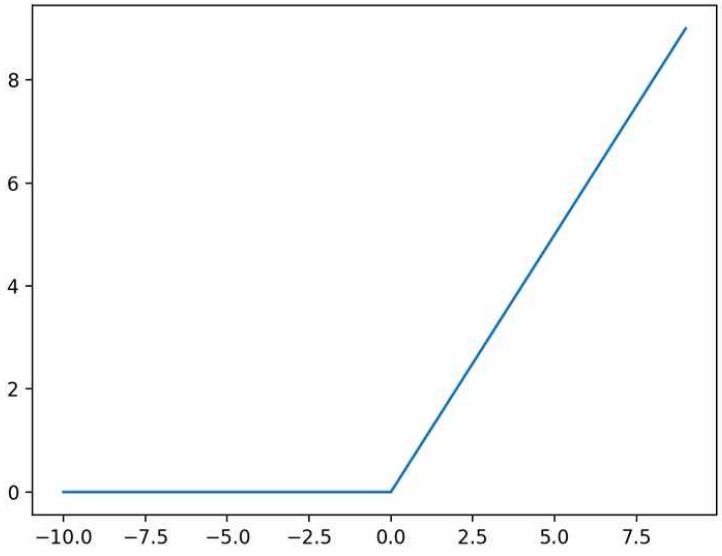


Figure 16. ReLU function according to z

2.4.6 Deep Neural Network (DNN)

The deep neural network (DNN) is an advanced form of the neural network. DNN is a deep network structure that increases the number of hidden layers in the existing multi-layer perceptron structure, as shown in Figure 17.

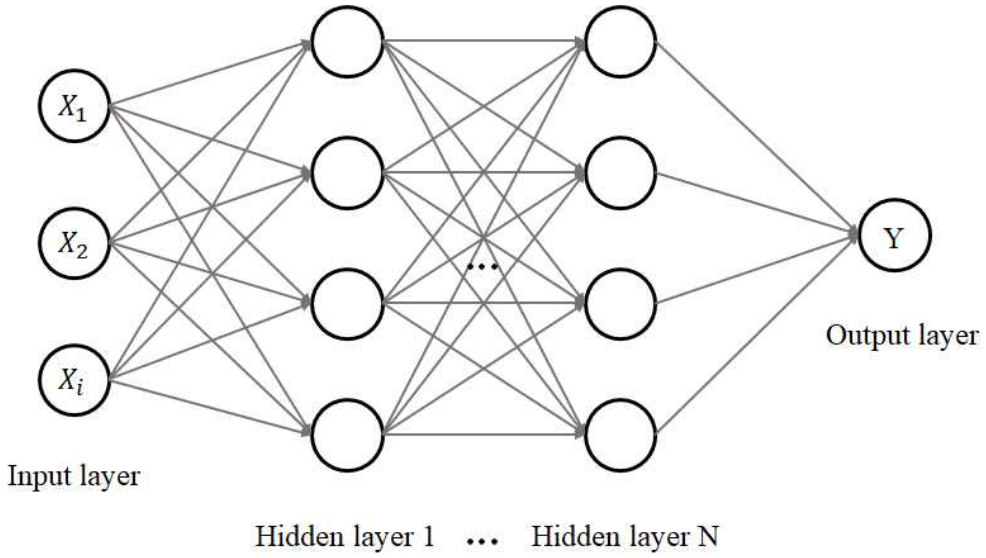


Figure 17. The structure of a deep neural network

The classification method of DNN is a supervised learning method that compares the output value (predicted value) through the forward propagation of the DNN with the value of the actual target, calculates the error, and corrects the weight of the hidden layer through backpropagation. The error is expressed as an error function and can be expressed as (6). When the result of the error function E is closer to 0, the neural network correctly predicts the outcome.

$$E = \frac{1}{2}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{6}$$

Thus, the training data is used for learning by repeating the forward propagation and backpropagation processes, and the performance is determined by checking the classification results from the test data.

2.5 Deep Learning

2.5.1 Convolutional Neural Network (CNN)

Unlike RNN, CNN is a hierarchical model that repeatedly performs convolution and pooling operations on the input data of multi-dimension images and performs classification through a complete connection operation. The networks consist of input layers, convolutional layers, a fully connected layer, and a softmax layer in order. In [43], CNN was used by LeCun to recognize handwritten numbers from 0 to 9. LeNet has three convolutional layers (C1, C3, C5), two minimum pooling layers (S2, S4), and a fully connected layer (F6) as shown in Figure 18.

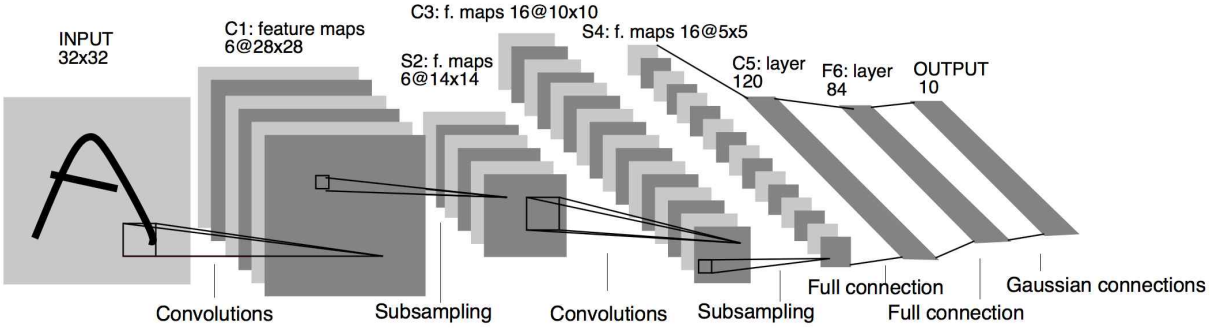


Figure 18. The structure of LeNet

(1) Convolutional layer

The convolutional layer is a layer in which an input image is two-dimensionally filtered through a convolution operation and transmitted to the next layer. The features from the input image are extracted using a convolution kernel (or a filter). Before CNN was widely used, users designed filters to extract features for learning, and the learning performance significantly changed according to the number and features of the filters. To eliminate such pre-processing, the convolutional layer of

CNN uses a method of initializing the filter values to arbitrary values and then induces the filter values to be optimal through learning. In (7), an environment is expressed, in which the size of the l-th convolutional layer is $(N-m+1)\times(N-m+1)$ when an $N\times N$ input image is connected to a convolutional layer and an $(m\times m)$ sized convolutional kernel is used.

$$x_{ij}^{(l)} = \sum_{u=0}^{m-1}\sum_{v=0}^{m-1} x_{(i+u)(j+v)}^{(l-1)} k_{uv}^{(l)} + b_i^{(l)} \quad \text{for} \quad i,j = 0,1,\ldots,N-m \tag{7}$$

where, $k_{uv}^{(l)}$ is the kernel connected to the uv-th node of the l-th layer, and b is the bias. Figure 19 shows the convolution operation.

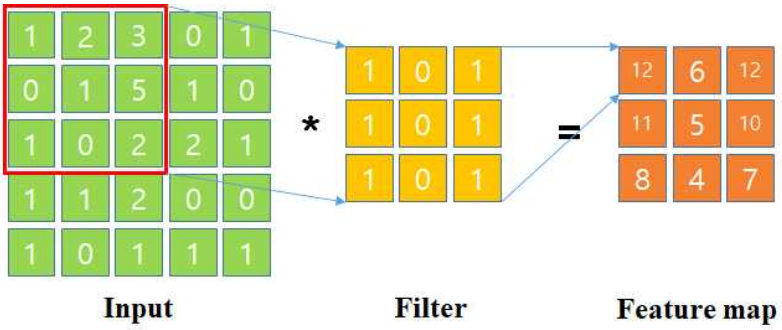


Figure 19. The example of convolution operation

(2) Pooling layer

The pooling layer is used to remove redundant feature values or highly correlated components of maps generated as a result of overlapping kernel operations of the convolutional layer. For example, if a 2×2 pooling patch is used, the number of nodes existing in the previous convolutional layer is reduced to 1/4 size and transferred to the next layer. Currently, the pooling layer used in CNN structures

includes a maximum pooling layer that delivers the maximum value of a node, an average pooling layer that delivers an average value, and a minimum pooling layer that delivers the minimum value. Figure 20 shows the max pooling operation.

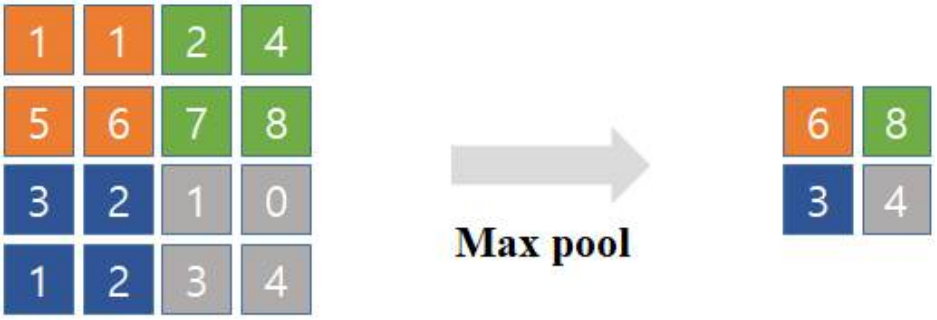


Figure 20. The example of max pooling operation

(3) Softmax layer

The softmax layer is a method of calculating the loss value for the probability distribution of each class output from the last layer. CNN learns according to the change of the calculated loss value. The method expressed in (8) calculates the soft-max probability distribution for each class, and (9) is a method of calculating the loss value of the education distribution.

$$p_k = \frac{e^{x_k}}{\sum_{i=1}^{N} e^{x_i}} \tag{8}$$

$$Loss = -\frac{1}{N}\sum_{n=1}^{N} \log(p_n, l_n) \tag{9}$$

N is the number of all classes to be classified, p_n is the probability of the n-th class output from the CNN, and l_n is the target value of the n-th class.

2.5.2 Recurrent Neural Network (RNN)

An RNN model is neural network architecture using a single layer or multiple layers, consisting of circulation connections, commonly applied for learning the data of temporal or sequential, like voice, video, and string. This network model has the characteristic of memorizing the state of previous information, then applying it to the current input data. This mechanism of the RNN model has an advantage in learning the sequential data.

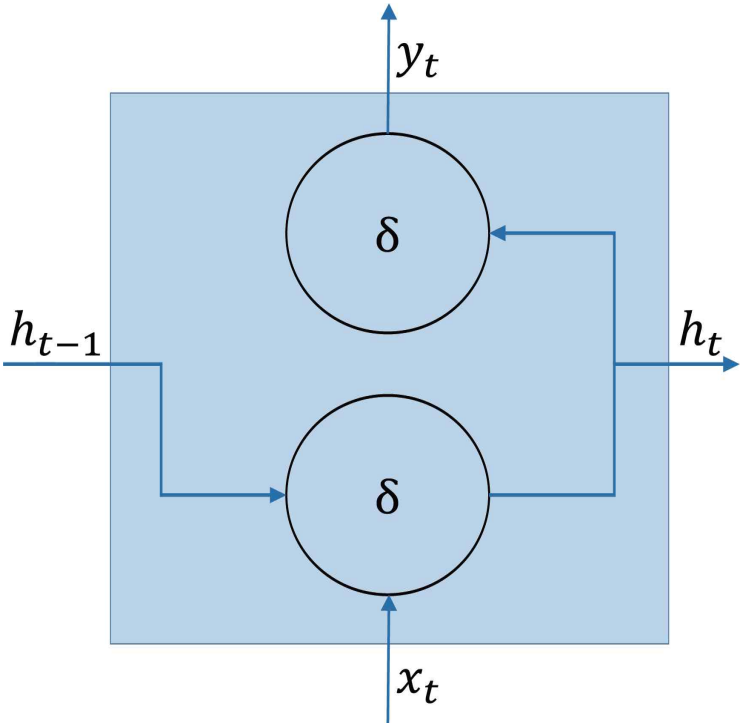


Figure 21. Schematic of an RNN node [44]

As shown in Figure 21, an RNN node consists of the current input x_t, output y_t, previous hidden state h_{t-1}, and current hidden state h_t. The current hidden state is determined as,

$$h_t = \delta_{hidden}(W_{hidden}h_{t-1} + W_{input}x_t + b_{hidden}) \quad (10)$$

$$y_t = \delta_{output}(W_{output} h_t + b_{output}), \tag{11}$$

where δ_{hidden} and δ_{output} are the functions of activation, which output to hidden and output layer, respectively. W_{input}, W_{output} and W_{hidden} are weights of recurrent connection between the input-to-hidden, hidden-to-output, and hidden-to-hidden layers, respectively. b_{hidden} and b_{output} are the bias terms for the hidden state and output state, respectively. Here, the function of activation has element-wise linearity or non-linearity characteristic, selected from several existing functions such as the sigmoid, rectified linear unit, or hyperbolic tangent.

2.5.3 Long Short-Term Memory (LSTM)

In a conventional RNN model, it can be hard to train the lengthy sequential data because of divergence or vanishing gradient problems that disturb the ability of the network to backpropagate gradients like a long-term dependency problem [45]. The LSTM-based RNN model replaces the conventional RNN node with LSTM to solve the long-term dependency problem in the learning data. LSTM includes memory blocks withal memory cells called 'gates' on the repetitive hidden layer, as depicted in Figure 22. On the memory cells, the gates control the states of new information forgetting and updating the previous hidden states, and decision the output. The function of component of each cell is as follows:

- Input gate (i_t) controls the input activation of new information into the memory cell.
- Output gate (o_t) controls the output flow.
- Forget gate (f_t) controls when to forget the internal state information.
- Input modulation gate (g_t) controls the main input to the memory cell.
- Internal state (c_t) controls the internal recurrence of cell.
- Hidden state (h_t) controls the information from the previous data sample within the context window:

$$i_t = \delta(U_i x_t + W_i h_{t-1} + b_i) \tag{12}$$

$$o_t = \delta(U_o x_t + W_o h_{t-1} + b_o) \tag{13}$$

$$f_t = \delta(U_f x_t + W_f h_{t-1} + b_f) \tag{14}$$

$$g_t = \delta(U_g x_t + W_g h_{t-1} + b_f) \tag{15}$$

$$c_t = f_t c_{t-1} + g_t i_t \tag{16}$$

$$h_t = \tanh(c_t)o_t, \tag{17}$$

where the U and W terms are weight matrices and b terms are bias vectors. When the LSTM-RNN model trains a dataset for learning, that model focuses on learning the parameters b, U, and W of the cell gates, as expressed in (12)-(15).

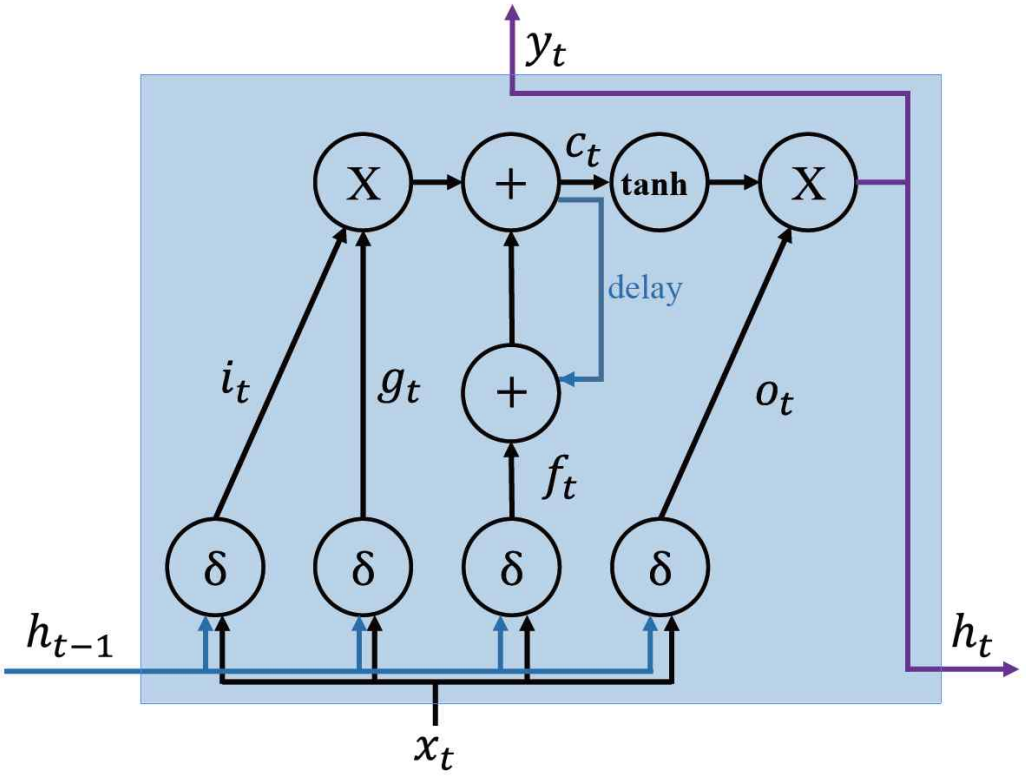


Figure 22. Schematic of an LSTM memory cell structure with an inner recurrence c_t and an outer recurrence h_t, i_t, o_t, f_t, and g_t.

2.6 Performance Metrics

To prove the performance of deep learning models, it used four evaluation metrics getting multi-class classification [46].

1. Precision: it calculates the number of identifications of the true person (person A, B, ... G) out of the positive classified classes. The average precision of each individual class (POC: the precision of each individual class) means to the overall precision (OP):

$$Per-POC_c = \frac{tp_c}{tp_c + fp_c} \tag{18}$$

$$OP = \frac{1}{C}\left(\sum_{C=1}^{C}\left(\frac{tp_c}{tp_c + fp_c}\right)\right), \tag{19}$$

where tp_c is the true positive rate of a person classification (c =1, 2,..., c), fp_c is the false positive rate, and C is the number of classes in the dataset.

2. Recall (Sensitivity): it calculates the number of persons correctly classified out of the total samples in a class. The average recalls for each class (RFC: recalls for each class) means the overall recall (OR):

$$Per-RFC_c = \frac{tp_c}{tp_c + fn_c} \tag{20}$$

$$OR = \frac{1}{C}\left(\sum_{c=1}^{C}\left(\frac{tp_c}{tp_c + fn_c}\right)\right), \tag{21}$$

where fn_c is the false negative rate of a class c.

3. Accuracy: it calculates the proportion of correctly predicted labels (the label is the unique name of an object) about overall predictions. It means an overall accuracy (OA):

$$OA = \frac{TP + TN}{TP + TN + FP + FN}, \tag{22}$$

where, $TP = \sum_{c}^{C} tp_c$ is the overall true positive rate for a classifier on all classes, $TN = \sum_{c}^{C} tn_c$ is the overall true negative rate, $FP = \sum_{c}^{C} fp_c$ is the overall false positive rate, and $FN = \sum_{c}^{C} fn_c$ is the overall false negative rate.

4. F1-score: it is the weighted average of precision and recall.

$$F1 - score = \sum_{c=1}^{C} 2\left(\frac{n_c}{N}\right)\frac{precision_c \times recall_c}{precision_c + recall_c}, \tag{23}$$

where n_c is the number of samples in class c and $N = \sum_{c=1}^{C} n_c$ is the entire number of individual examples in a set of C classes.

2.7 Related work

2.7.1 Feature extraction

For noise reduction and feature extraction in ECG biometrics, diverse approaches have been discussed in the literatures. Particularly, Odinaka, I. et al. categorizations based on features and classifiers [19]. First of all, the categorized the feature extraction of ECG includes a feature extraction algorithm using fiducial, non-fiducial, and hybrid features. For a fiducial-based algorithm, temporal, amplitude, angular, or morphology features are extracted from ECG characteristic points. The extracted ECG features involve specific characteristics information such as the interval of each ECG wave of P, QRS, and T wave [20]. For a non-fiducial-based algorithm different from the fiducial-based algorithm, the direct features are not used. The indirect features are used, such as autocorrelation and wavelet coefficients [21-23]. Because such the indirect features extraction disuses the method of using ECG characteristic point, for making the composition of a feature set like a heartbeat segmentation and alignment, most methods use the method of R-peak detection. A part of the remainder methods has to detect the major constituent of ECG wave, such as the start end of the P wave, the start end of the QRS complex, and the start end of the T wave. An approach combining with the fiducial and non-fiducial is used for a hybrid feature extraction method. In addition, a classifier is required for the categorization, such as LDA, SVM, KNN, generative model, match score classifier, and neural networks. Consequentially, a fiducial, non-fiducial, and hybrid (characteristic point, similarity, a combination of the fiducial and non-fiducial) algorithm of feature extraction can be used for classifying the ECG.

2.7.2 Methodology

In biometrics, there are two basic recognition methods. The one is the authentication method. This method works on the one-to-one comparison with observed bio-signal with a template signal stored in a database to verify if the one's signal is exactly right or not. The other one is the identification method. This method carries out the one-to-many comparison with a database trying to establish the identity of an unknown human. If the result of a comparison between the template of the database and a specific bio-signal sufficient the criteria at fiducial, the individual identifying will succeed [24].

Various techniques have been proposed for ECG biometric systems using diverse ECG databases [25]. In [25], the authors have analyzed different studies in order to compare the authentication scenarios, identification equal error rates (EER), and averages of classification accuracy using pathological and normal signals ECG databases. According to their results, the overall EER was 0.92% in an authentication scenario, and the weighted average rate was 94.95% in an identification scenario. Their results in [25] revealed that the number of ECG leads used affects the recognition performance, and the choice of features influences the accuracy rate of identification.

In plenty of recent studies, ECG biometrics has applied deep learning methods to solve the problem [26-34]. In [30], it has been used for a convolutional neural network (CNN) to classify patient-specific ECG heartbeats. In [31], it designed a residual convolutional neural network (ResNet) applying a mechanism of attention for ECG authentication of humans. For a recurrent neural network (RNN), RNN has merit different from CNN when processing 1-D signals like ECG. Typically, CNN processes 2-D data including more than 2 × 2 signals or an image for object classification and identification. RNN processes continuous or sequential 1-D data, such as a sensor signal, voice, and text for classification and identification. For instance, RNN has been applied to classify the form of an ECG beat in [32].

However, because a conventional RNN occurs vanishing gradients, training the network is difficult using long-term sequences of data. To resolve this problem, a long short-term memory cell (LSTM) and gated recurrent units (GRU) have been studied and proposed (The GRU has improved LSTM in an aspect of training speed) [35,36]. Because of overcoming the vanishing gradients and representing good performance, the RNN model with LSTM has been widely applied in various applications, such as ECG biometrics, handwriting recognition, speech recognition, and etc [37,38]. Furthermore, the dropout technique can be utilized for reducing overfitting in the deep learning system [39]. Overfitting is verified if the model of deep learning carries out well while using its training dataset and it carries out badly while using its testing dataset. In [40], LSTM in ECG biometrics proved to be more appropriate than GRU for classification and identification. Therefore, the LSTM-based RNN model was applied to authenticate and identify problems using ECG data [40-42]; the techniques of deep learning have shown much stronger performance compared to other non-deep learning techniques.

2.7.3 Recent work

As mentioned in Chapter 2. 7. 1 and 2. 7. 2, the ECG-based biometrics can be represented into two main applications based on classification, identification and authentication using the three feature extraction method. Although different techniques have been proposed for classification, identification, or authentication, it is difficult to compare and analyze the methods because of various datasets applied in their experiments.

R. Salloum et al. [40] evaluated the conventional RNN, LSTM-based RNN, and GRU-based RNN model using two publicly available datasets: ECG-ID Database (ECGID) and MIT-BIH Arrhythmia Database (MITDB). The experimental results of [40] showed that their LSTM-based RNN model with a unidirectional feature, resulted in a classification accuracy of 100%. However, when their input size was short (input sequence length is 3), the classification accuracy is decreased. A more recent paper [29] evaluated GRU-based RNN methods using two publicly available datasets: ECGID and MITDB. The experimental results of [29] showed that their proposed model, the combination of GRU-based RNN with a bidirectional feature, resulted in a classification accuracy of 98.55%.

In the study by Ö. Yildirim [67], the author classified the five types of heartbeats in MITDB. The results of the proposed model experiments showed a recognition performance of 99.39%. In [33], the authors evaluated the identification rate of eight datasets using a proposed multiresolution CNN. Their results verified an average identification rate of 96.5% and 93.5% in normal and abnormal ECG datasets, respectively. Additionally, X.Zhang [66] evaluated conventional methods using a private dataset of recordings from 91 subjects. Further, they applied two types of input data using the ECG and ballistocardiogram (BCG) for identification

and authentication with RNN, LSTM, and GRU. When using BCG, the identification accuracy was 97.8%, whereas the accuracy of ECG was 98.9% for identification. For the ECG biometric, different deep learning techniques, because of high performance, are used in various applications, including handwriting recognition, phoneme classification, image recognition, voice recognition, automatic driving, etc.

Ⅲ Proposed Deep RNN Method and Preprocessing Procedures

3.1 Experiment configuration and flow

This thesis configured and performed ECG recognition experiments for the personal identification to improve the performance, as shown in Figure 23.

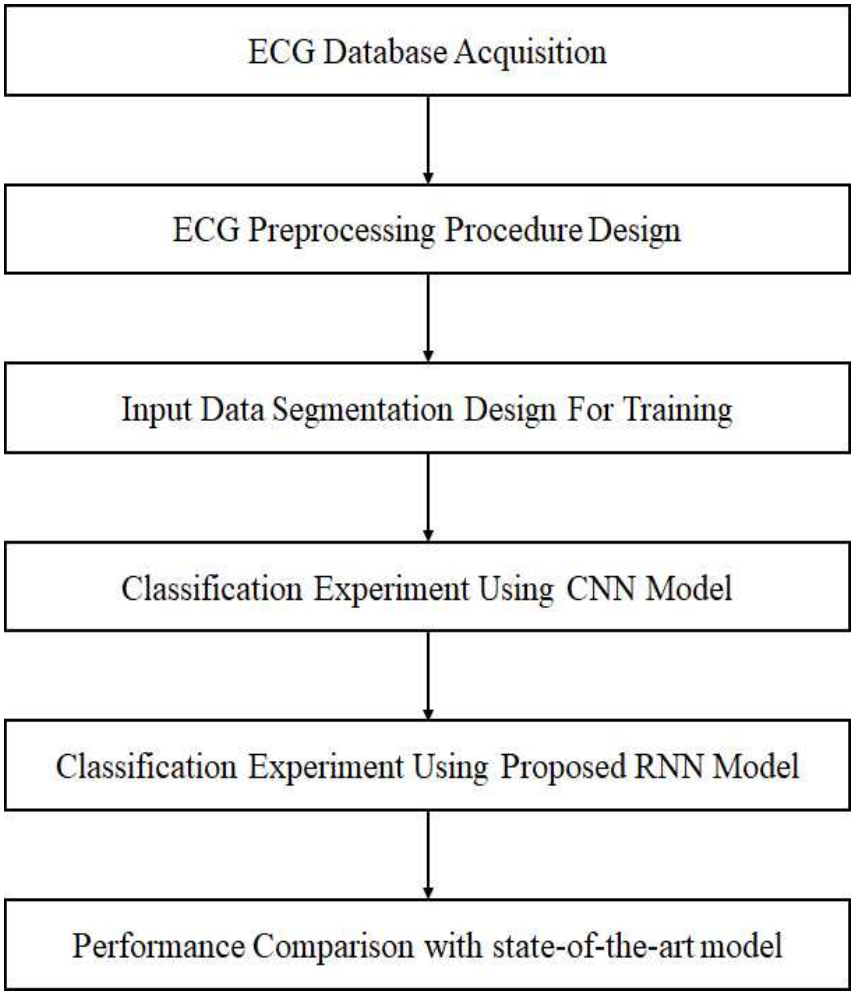


Figure 23. Flowchart of ECG recognition performance of experiment Using CNN and RNN

First, it prepared a public ECG dataset to be used in the experiment. A detailed description of the data set used in this thesis is given in Section 3.5. To improve ECG identification performance, the ECG training dataset was constructed by creating data to which the ECG preprocessing and data segmentation method were applied and training data without data segmentation. Subsequently, the CNN model, RNN model, and the proposed RNN model for ECG identification were designed, and training and testing were performed. The designed learning models compared the accuracy according to the learning while learning the data set to which the preprocessing process and data segmentation method were applied. When training is complete, the test data is used to verify the test accuracy. The performance of the model used in the experiment is accurately checked through performance metrics. Additionally, this thesis evaluated whether the proposed model has improved performance by comparison with the state-of-the-art model.

3.2 Proposed Deep RNN Method

Figure 24 showed a schematic in the ECG identification system using the proposed Deep RNN method. It carried out a direct mapping from personal ECG inputs to personal label classification. When classifying the personal labels uses a specific time window.

The input is divided into a discrete sequence of equally spaced samples ($x_{1,}x_{2,}$, ..., x_t), where each data point x_t is a vector of the personal ECG signal. The samples are passed to an LSTM-based RNN model after being segmented by the window of size T, consisting of n segmented ECG signal components with a period of P.

The classification accuracy is low in the conventional RNN and LSTM-based RNN models if less than nine of the ECG groups are used for training and testing [40]. This thesis used three, six, and nine ECG groups (n = 3, 6, 9). The outputs receive a score by denoting the personal label prediction at each time step (y_1^L, y_2^L, ..., y_k^L), where $y_k^L \in R^c$ is a vector of classification scores representing the given input group, L is for layer, and c is the number of person classes. The score is computed at each time step for the personal label at time t. The multi-prediction for the whole window T is obtained by sum all the scores into one prediction.

For classification, this thesis used a late-fusion technique, called ``sum rule'', which is theoretically discussed in [47,48].

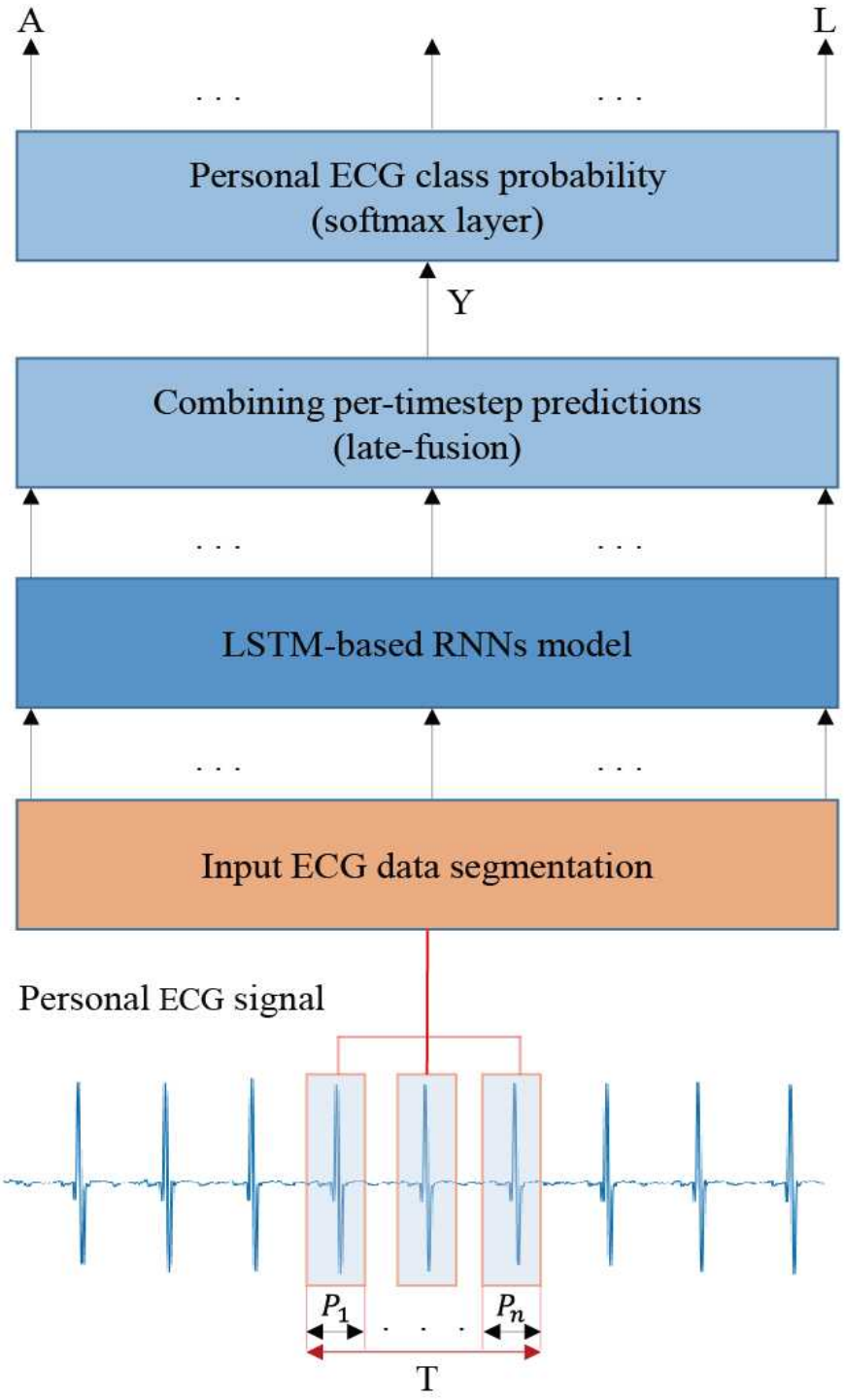


Figure 24. Proposed ECG identification architecture using LSTM-based RNN Model. The inputs are preprocessed raw signals from datasets, segmented into ECG components with a window size of T, and trained at the LSTM-based RNN model.

To convert the prediction scores to probabilities, A softmax layer on Y of the prediction score is applied.

$$Y = \frac{1}{T}\sum_{t=1}^{T} y_t^L \tag{24}$$

This thesis uses a bidirectional LSTM-based DRNN model for additional performance enhancement, as presented in Figure 25 [49].

It contains two parallel LSTM tracks: forward and backward loops for utilizing the context from the past and future of a specific time step to predict its label [42,50]. At each layer, there is a forward track ($LSTM^{fl}$) and backward track ($LSTM^{bl}$). The two tracks read respectively the ECG input from left to right and from right to left:

$$y_t^{fl}, h_t^{fl}, c_t^{fl} = LSTM^{fl}(c_{t-1}^{fl}, h_{t-1}^{fl}, x_t; W^{fl}) \tag{25}$$

$$y_t^{bl}, h_t^{bl}, c_t^{bl} = LSTM^{bl}(c_{t-1}^{bl}, h_{t-1}^{bl}, x_t; W^{bl}) \tag{26}$$

where y_t^{fl} and y_t^{bl} are the output of the prediction, h_t^{fl} and h_t^{bl} are the output of the hidden layer, and c_t^{fl} and c_t^{bl} are respective the present output in the forward and backward layers (l=1, 2, ..., L). The top layer L is the output of the sequence score from the forward LSTM and backward LSTM at each time step. Hence, the combined scores $Y \in R^c$ represent the prediction score of a person label. In this particular case, the late-fusion is expressed as follows:

$$Y = \frac{1}{T}\sum_{t=1}^{T} (y_t^{fL} + y_t^{bL}) \tag{24}$$

To evaluate the performance of the proposed model, the experiments of ECG

classification with six RNN structures are carried out, as shown in Table 3. This thesis chooses Arch 6 as the best model through the experiments because it results in the best classification performance.

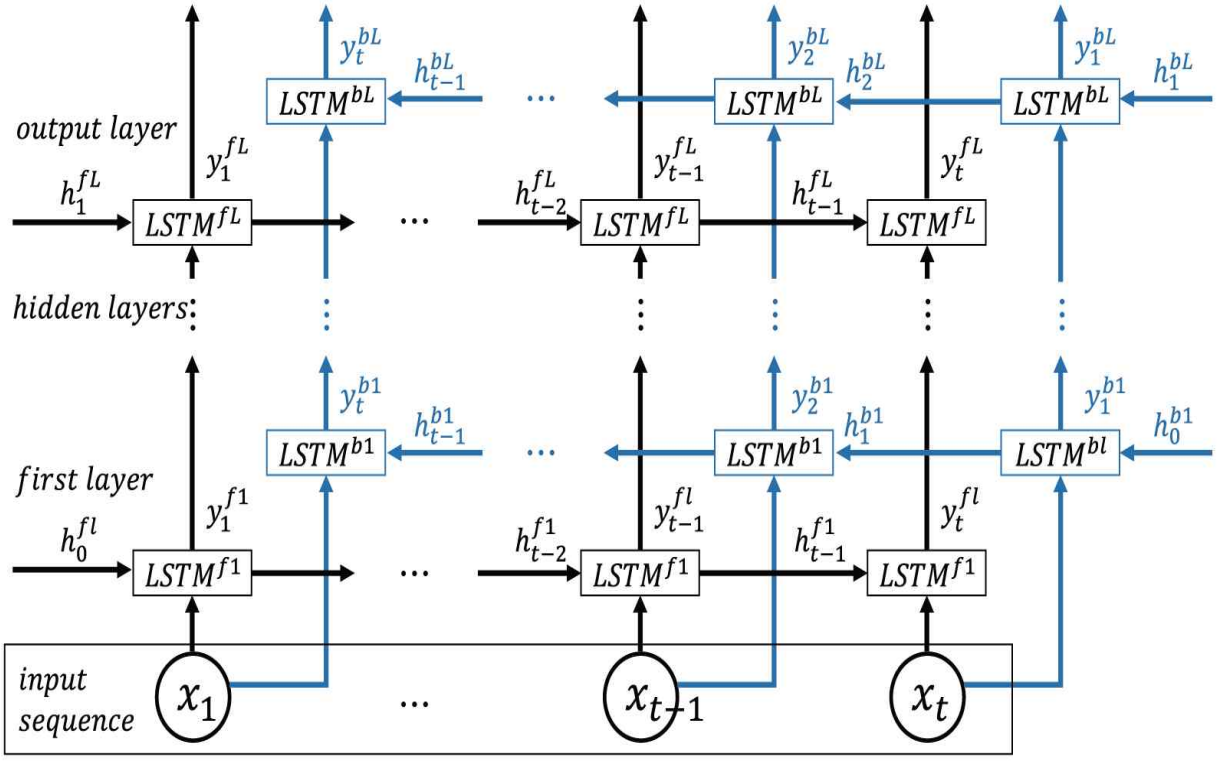


Figure 25. Bidirectional LSTM-based DRNN model consisting of an input layer, multiple hidden layers, and an output layer with forward $LSTM^{fl}$ and backward $LSTM^{bl}$ tracks [42].

In detail, Arch 1, 2, and 3 consist of unidirectional networks. In Arch 1, 2, and 3, the number of hidden layers is 1, 2, and 3, respectively. Moreover, Arch 4, 5, and 6 for the proposed networks model consist of bidirectional networks including the late-fusion layer, and the number of hidden layers is 1, 2, and 3, respectively.

Table 3. conventional LSTM (LSTM) and proposed bidirectional LSTM (pLSTM) architectures.

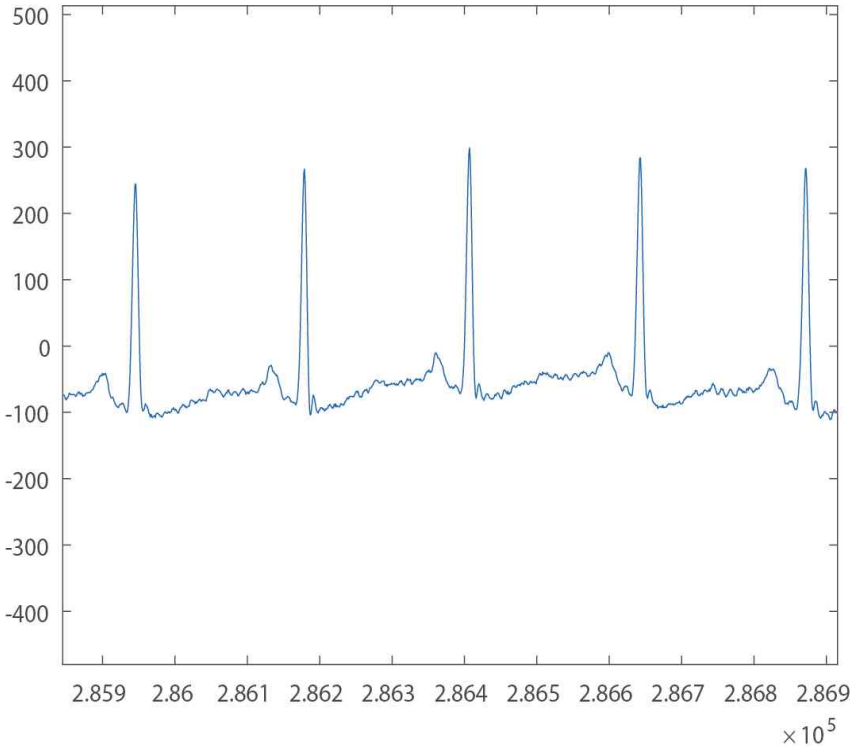
Architectures	Layers Type
Arch 1	LSTM-softmax
Arch 2	LSTM-LSTM-softmax
Arch 3	LSTM-LSTM-LSTM-softmax
Arch 4	pLSTM-late-fusion-softmax
Arch 5	pLSTM-pLSTM-late-fusion-softmax
Arch 6	pLSTM-pLSTM-pLSTM-late-fusion-softmax

3.3 Proposed Preprocessing Procedure

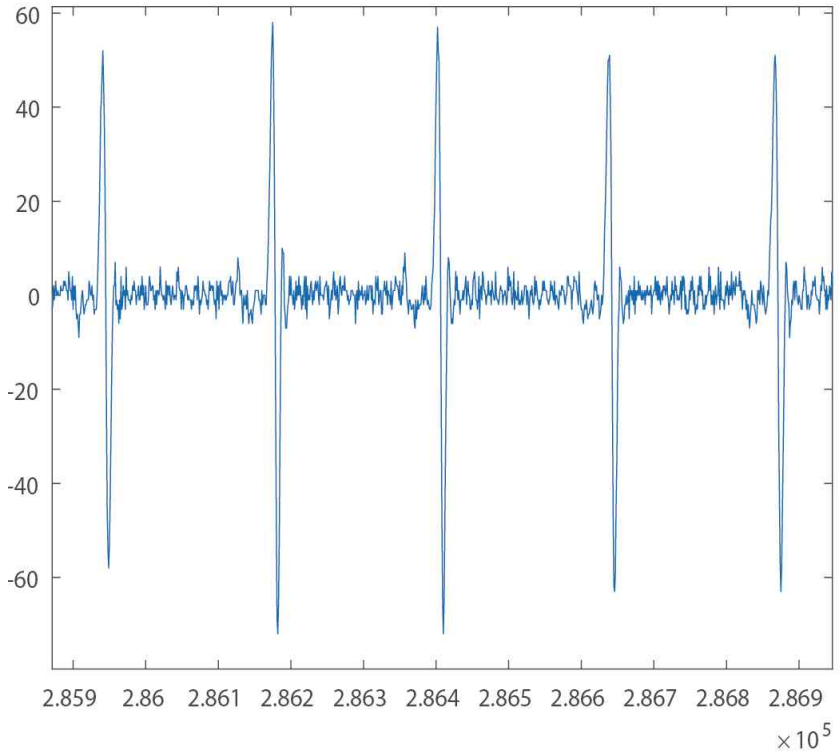
In this thesis, the ECG database used is obtained from the publicly available MIT-BIH Normal Sinus Rhythm (NSRDB) and MIT-BIH Arrhythmia datasets (MITDB), which are the portion of the Physionet database [51-53]. For the analysis, it carried out the preprocessing and segmentation of each dataset. Given an ECG recording, the proposed preprocessing procedure is applied in the first step. This procedure consists of applying the derivative filter, moving average filter, and amplitude normalization using (25) in the given order, as shown in Figure 26. The derivative filter is used to detect the more precise variation. For an average moving filter, it is used to reduce the noise resulting from the derivative filter. Then, the normalization is applied to learn more accurately in the training phase.

$$y[n] = 2(x[n] - x_{median})/(x_{\max} - x_{\min}), \tag{25}$$

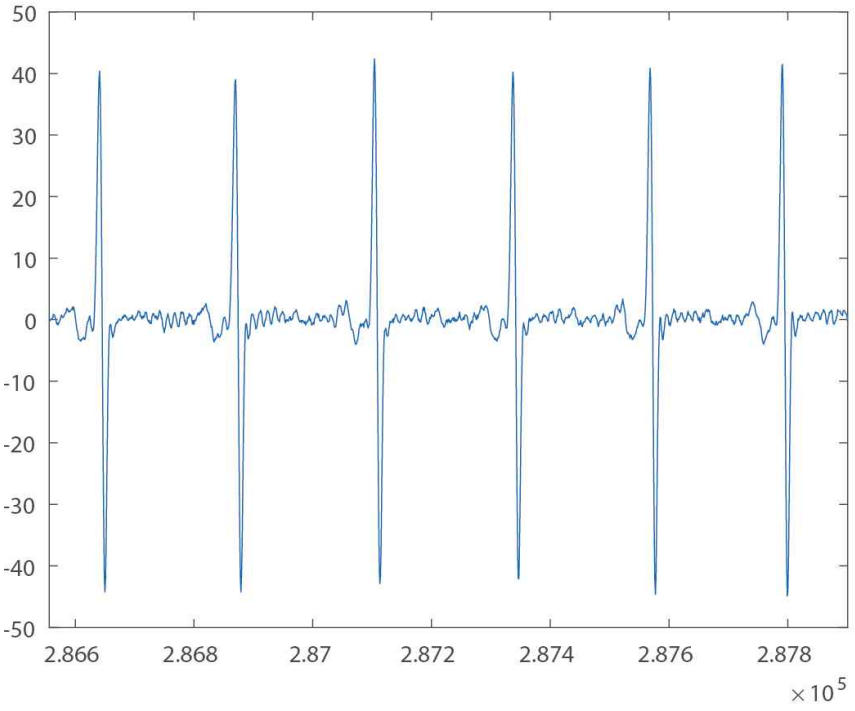
where $x[n]$ is the n-th value, x_{median} is the median value, $x_{\max}$ is the maximum value, and $x_{\min}$ is the minimum value of the input signal [54]. The next step is to segment the ECG recordings into ECG signal components with a period of P. The conventional segmentation technique uses an R peak as a marker from the segmented individual heartbeat waveforms: P wave, QRS complex, and T wave. For the NSRDB, 288 samples were trimmed and grouped, while for the MITDB, 444 samples were trimmed and grouped.



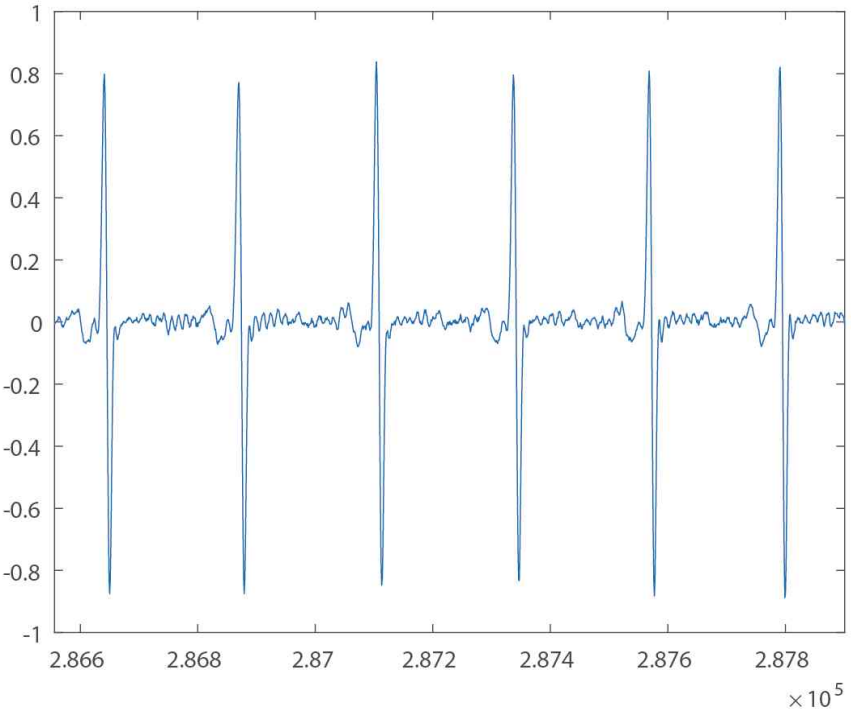
(a)



(b)



(c)



(d)

Figure 26. ECG signal preprocessing before the segmentation for input data: (a) ECG raw signal; (b) signal obtained after derivative filter; (c) signal obtained after moving average filter; (d) signal obtained after normalization.

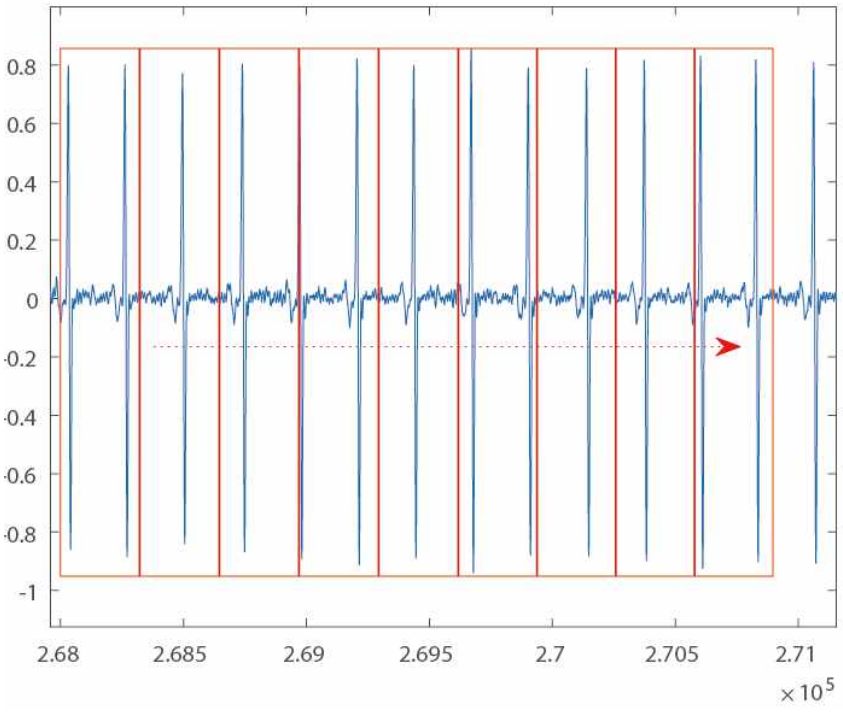
3.4 Classification Procedure

In the classification procedure, each ECG dataset is divided into a training and testing set. Each training or testing sequence is of one $\times$ N size, where N is the number of samples in the ECG signal. After one-hot sequences encoding, the weight parameters of the bidirectional LSTM are determined using the training set [55]. Then, the softmax function is used to obtain a class probability (a set of the subject probability distribution). After the RNN training, the test sequence is fed to evaluate the RNN model. A classification decision for each test sequence is obtained by selecting the class with the highest probability in all classes.

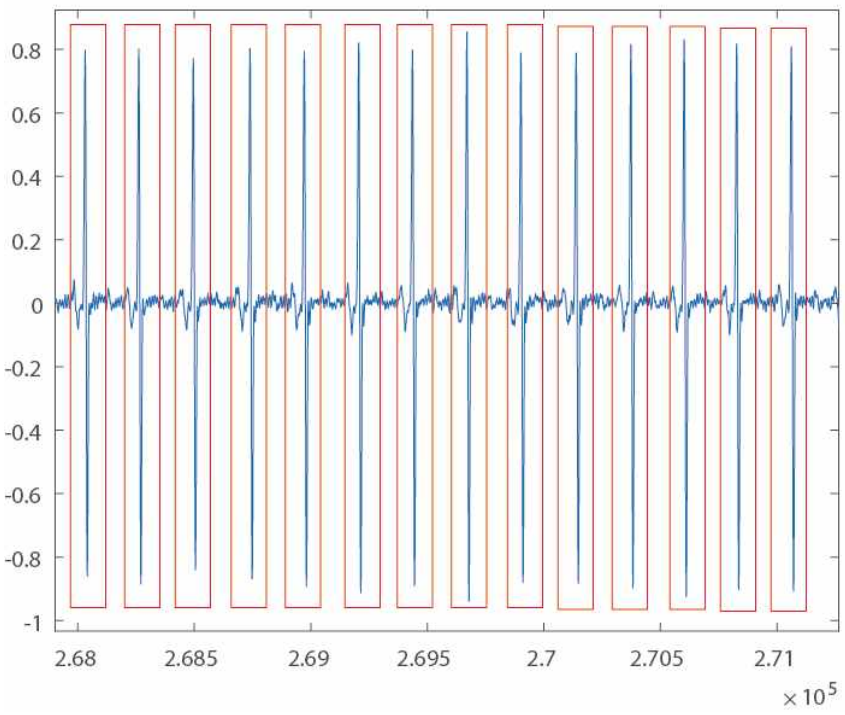
3.5 Dataset and Implementation

The NSRDB contains 18 two-channel recordings obtained from 18 subjects (5 males aged 26 – 45 and 13 females aged 20 – 50). Similarly, MITDB contains 48 two-channel recordings obtained from 47 subjects (25 males and 22 females). One recording for each subject was used in the proposed deep learning system. The recordings of the NSRDB were digitized using 12 bits per sample. Moreover, the recordings of the MITDB were digitized using an 11-bit resolution over a 10 mV range.

In the proposed method, the NSRDB and MITDB were applied in the segmentation process using the sampling frequency of the dataset. Here, the NSRDB and MITDB can be segmented using a fixed segmentation time-period or conventional R-peak detection owing to the irregular ECG waveform [56]. To apply the real-time system, it considered the smallest input data size concerning the minimum R-R interval. According to the clinical definition, the minimum R-R interval of 200 ms cannot exceed 300 bpm [57-59]. Thus, the selected NSRDB input size equals the time required for 288 samples (2.25 s), and the MITDB input size corresponds to the time required for 444 samples (1.23 s). Because it used the non-feature extraction method in the first and second experiment, the segmented data in NSRDB randomly included two to four heartbeats. The segmented data in MITDB randomly had zero to two heartbeats, as shown in Figure 27 (a). The third experiments used ECG signals segmented with R-peak detection, as shown in Figure 27 (b). After the R-peak detection, information on the ECG wave is obtained, such as the part of only including noise with the flow of time. Subsequently, based on R-peak, one ECG waveform consisting of 100 samples is extracted, including 50 samples to the left and 50 to the right.



(a)



(b)

Figure 27. ECG signal segmentation after the normalization: (a) a method using a fixed segmentation time period; (b) a method using R-peak detection.

For the ECG preprocessing, Matlab is used to manage the generation of training and testing data. The implementation, training, and testing of the RNN models were performed using TensorFlow [60]. The ECG identification system used the configuration and framework listed in Table 4. The tests were performed on the proposed model after the completion of every training epoch. The processed raw data is divided into two sets: 80% and 20% for the training and testing, respectively. The cost function used is the cross-entropy error during training, and the optimization method used is the Adam algorithm with a learning rate of 0.001 [61]. Experiment 1 was performed with a batch size of 1000, whereas experiments 2 and 3 were performed with a batch size of 100. The model parameters of the conventional and proposed LSTM are listed in Table 5. These parameters were selected through iterative experiments using these parameters. The different conditions of the evaluation were the number of layers, number of hidden units, and input sequence length. In terms of the learning time, 4, 8, and 16 h were required for the 1, 2, and 3 hidden layers, respectively.

Table 4. Server system configuration and framework

Category	Tools
CPU	Intel i7-6700k @ 4.00 GHz
GPU	NVIDIA GeForce GTX 1070 @ 8GB
RAM	DDR4 @24GB
Operating System	Windows 10 Enterprise
Language	Python 3.5
Library	Google Tensorflow 1.6 CUDA Toolkit 9.0/NVIDIA cuDNN v7.0

Table 5. Model parameters of conventional and proposed LSTM.

Parameter	Value
Loss Function	Cross-entropy
Optimizer	Adam
Dropout	1
Learning Rate	0.001
Number of hidden units	128 and 250
Mini-batch size	1000 and 100

Ⅳ Experimental Results and Discussion

4.1 Experimental Results

This thesis found various conventional classification methods being used on NSRDB and MITDB datasets. For the NSRDB dataset, the reported classification accuracy ranged from 99.4% to 100% [62,63], while for the MITDB dataset, the reported accuracy ranged from 93.1% to 100% [29,33,40,61-67]. The RNN-based method outperforms the aforementioned methods on both datasets. Also, to evaluate the proposed architecture about the late-fusion technique, First of all, the experiment performed with the basic CNN and the modified CNN with the late-fusion technique using only the NSRDB dataset.

For the NSRDB and MITDB datasets, the classification experiments were performed using one recording per subject; in the NSRDB experiment, ECG signal was segmented with a fixed segmentation time period, including 2-4 training and testing heartbeats per subject were used. Similarly, in the MITDB experiment, the unfixed group ECG including 0-2 training and testing heartbeats per subject were used. Moreover, ECG signals segmented with R-peak detection, including three, six, and nine training and testing heartbeats per subject were used. Because the sampling rate of the NSRDB and MITDB were different, the training and testing heartbeats per subject were set independently for a dataset.

4.1.1 Experimental Results of CNN

The experiment with CNN translates from the 1-D ECG dataset to the 2-D ECG dataset. The 1-D ECG data consist of 1 × M. To use the CNN model, the input ECG data was converted by N × N after adding the zero value, as shown in Figure 28. The input size of the NSRDAB dataset is 1 × 288 in this experiment. If these data is converted by 2-D, tt is not represented by a square matrix. The translated data after adding the zero value is considered a square matrix without data distortion in the CNN model. This is because the CNN model does not extract the zero value.

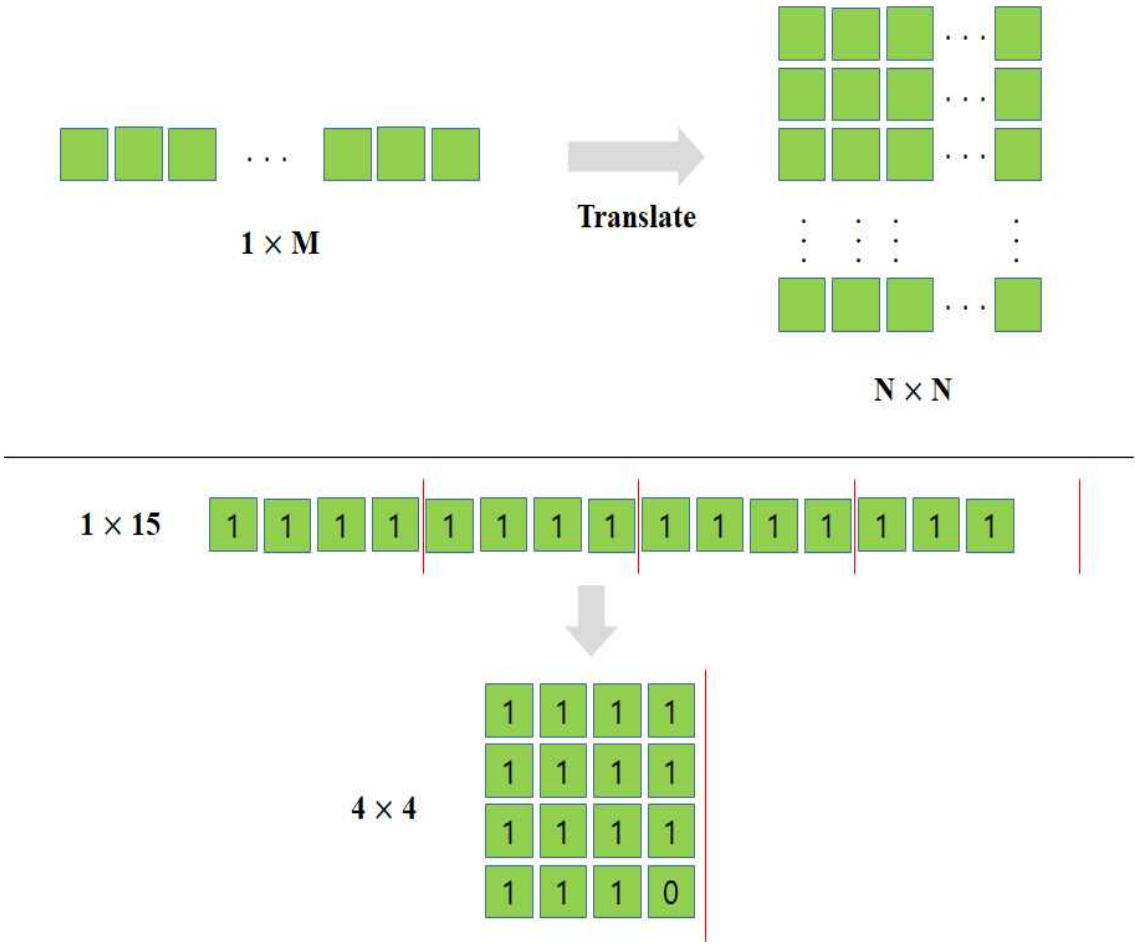


Figure 28. Example of ECG 2-D data convert

Table 6 shows the Basic CNN (bCNN) and modified CNN (mCNN) architectures. In detail, bCNN 1, 2, and 3 consist of the convolutional and softmax layers, and the number of hidden layers is 1, 2, and 3, respectively. Moreover, mCNN 4, 5, and 6 for the modified networks model consist of the convolutional layer and late-fusion, and the number of hidden layers is 1, 2, and 3, respectively.

Figure 29 shows the classification accuracy of the selected architectures and parameter conditions. In Figures 29, the number of hidden layers is 1, 2, 3, and a 2-D converted ECG signal segmented with a fixed segmentation time period was used. The input data form is 17 × 17.

The results of Figure 29 confirmed that the classification accuracy varied between 96.29 – 99.88%; the results presented were for different hidden layers and zero dropout. Thus, the modified CNN networks did not perform better than the conventional CNNs for the same experimental conditions. Consequently, the type of CNN model did not significantly improve the performance over existing models, although the deep models or late-fusion techniques are applied.

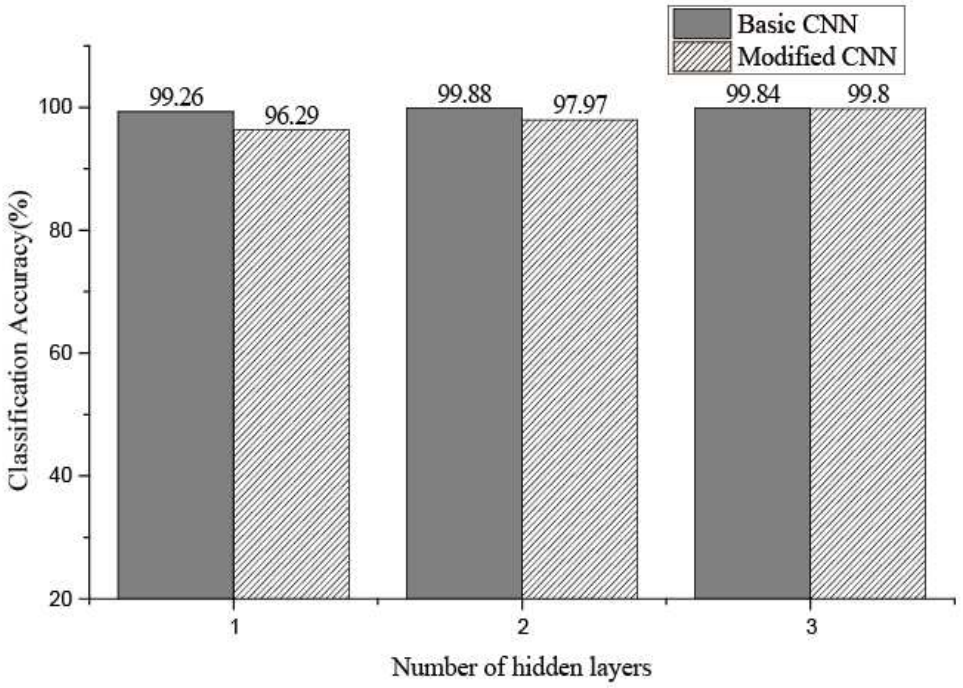


Figure 29. Classification accuracy for NSRDB. The input sequence length is 2-4 for the number of heartbeats, and input data form is 17 × 17.

Table 6. Basic CNN and modified CNN architectures (Convolution layer consist convolution , ReLu, and Pooling. “conv” means a convolution layer. “fc” means fully-connected layer).

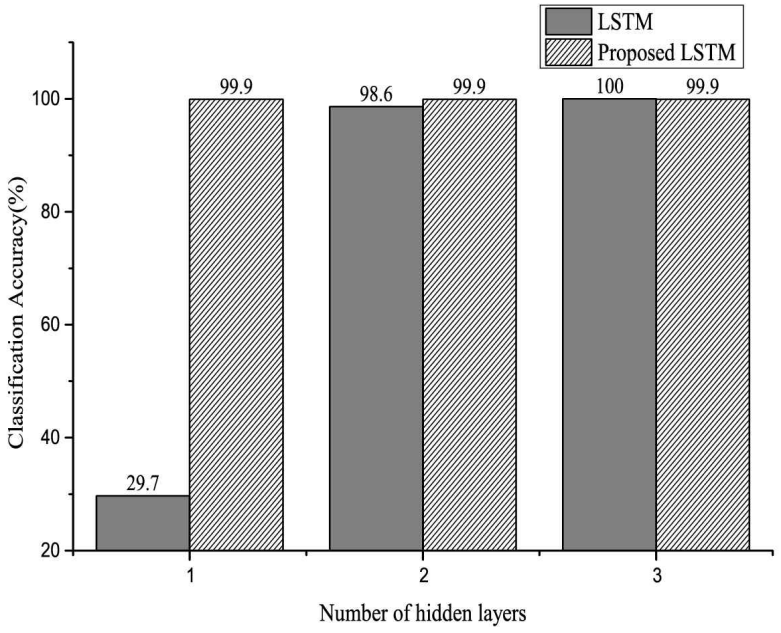
Architectures	Layers Type
bCNN 1	conv-fc-softmax
bCNN 2	conv-conv-fc-softmax
bCNN 3	conv-conv-conv-fc-softmax
mCNN 4	conv-fc-late-fusion-softmax
mCNN 5	conv-conv-fc-late-fusion-softmax
mCNN 6	conv-conv-conv-fc-late-fusion-softmax

4.1.2 Experimental Results of Proposed RNN

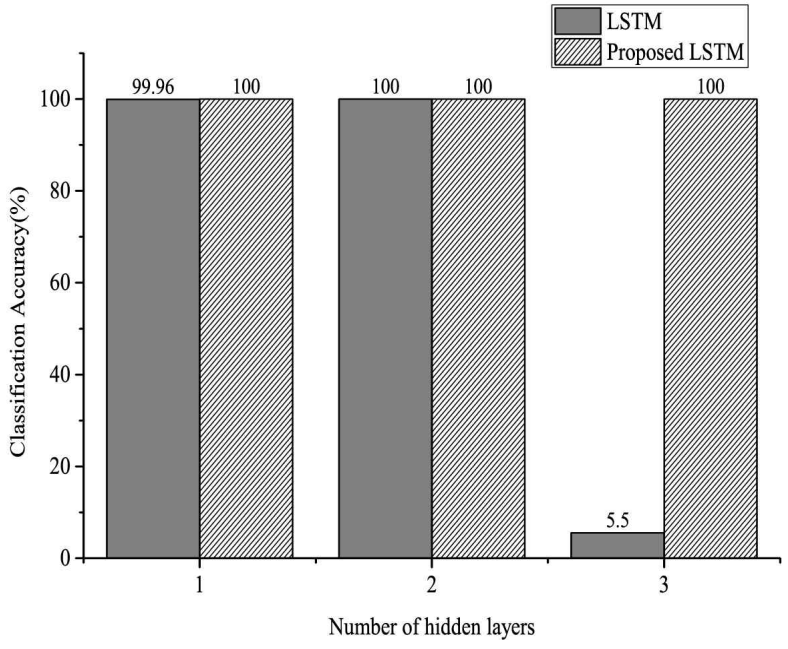
Figures 30 and 31 show the classification accuracy for the selected architectures and parameter conditions. In Figures 30 (a) and 31 (a), the number of hidden units of hidden layer is 128, and an ECG signal segmented with a fixed segmentation time period was used. The results of Figures 30 (a) and 31 (a) confirm that the classification accuracy varied between 29.7 – 100% and 1.87 – 98.53%, respectively. Furthermore, in the case of Figures 30 (b) and 31 (b), the number of hidden units of hidden layer is 250. The results of Figures 30 (b) and 31 (b) confirm that the classification accuracy varied between 5.5 – 100% and 2.21 – 99.73%, respectively. In Figure 32, the number of hidden units of the hidden layer is 250, and the ECG signal segmented with a fixed segmentation time period was used. Figure 32 confirms that the classification accuracy varied from 5.5 – 100% to 63.8 – 99.8%, respectively. Hence, the results presented are for different input sequence length, zero dropout, and number of hidden units. Thus, the proposed LSTM networks performed better than the conventional RNNs for the same experimental conditions. Furthermore, it can observe that a randomized decrease in the length of the input sequence—like the unfixed group ECG—improves the performance of the proposed LSTM networks and hyperparameter settings. In this experiments, the classification accuracy increased with a decrease in the number of hidden units and an increase in the number of hidden layers.

In [40], an increase in the number of the hidden layers and units increased the classification accuracy. However, in this experiments, the randomized short input sequence size—like the unfixed group ECG—resulted in an increase in the number of hidden layers and units and a decrease in the classification accuracy. Furthermore, as shown in Figure 32, the classification accuracy and number of

hidden layers increased when the input sequence group size was long.

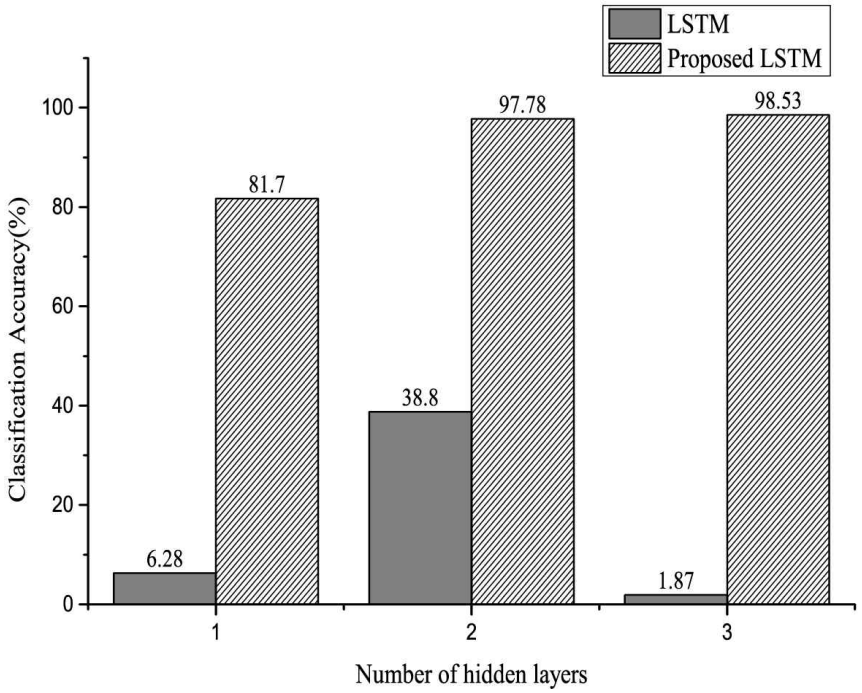


(a)

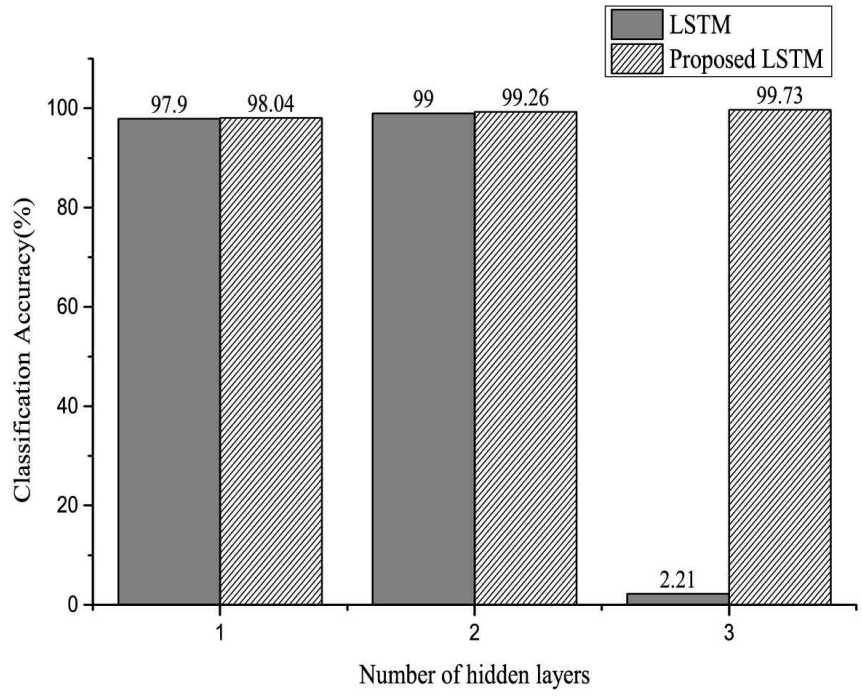


(b)

Figure 30. Classification accuracy for NSRDB using two selected parameters: the number of hidden unit of (a) is 128 and number of hidden unit of (b) is 250. The input sequence length is 2-4 for the number of heartbeats (Experiment 1).



(a)



(b)

Figure 31. Classification accuracy for MITDB using two selected parameters: the number of hidden units of (a) is 128 and number of hidden units of (b) is 250. The input sequence length is 0-2 for the number of heartbeats (Experiment 2).

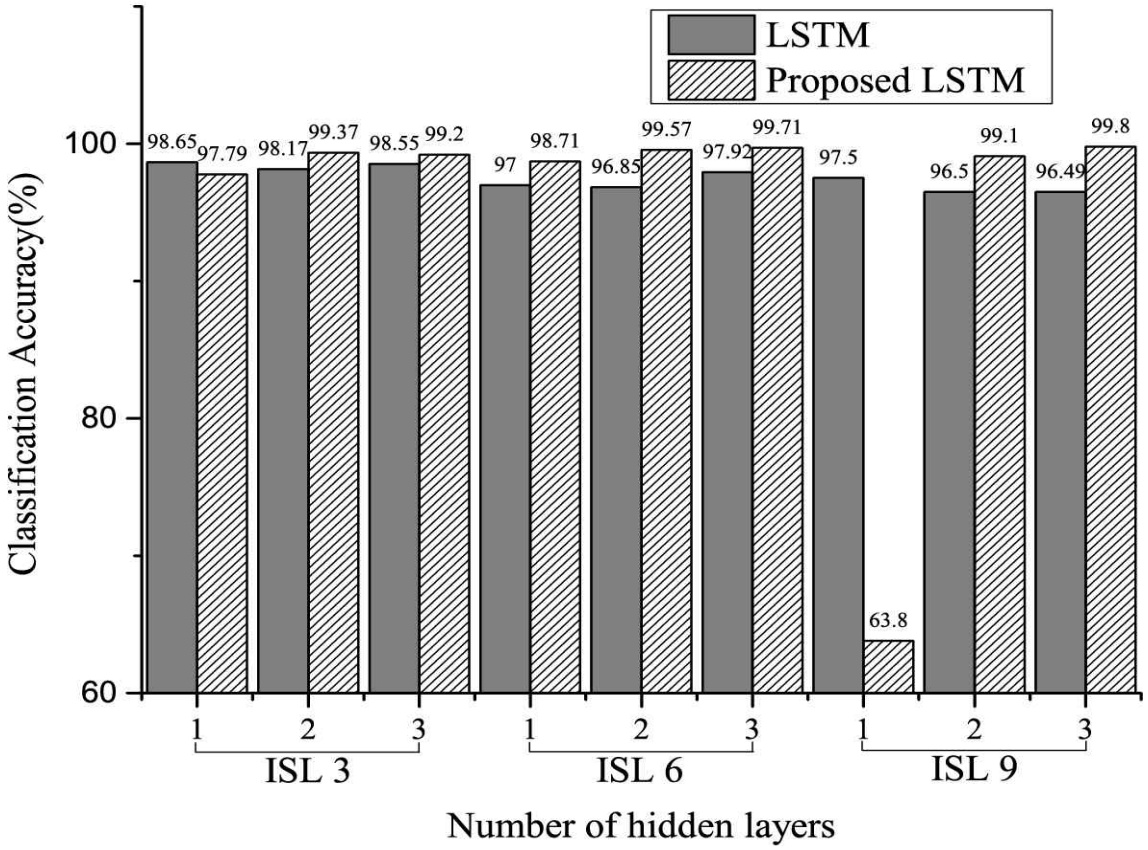


Figure 32. Classification accuracy for MITDB using selected parameters of 250 hidden units. The input sequence length (ISL) is 3, 6, and 9 for the number of heartbeats (Experiment 3).

Table 7. Performance summary of the proposed bidirectional LSTM in NSRDB analysis 1 of Figure 30 (a).

Type of Cell/Unit	Input Sequence Length (in Number of Beats)	Number of Hidden Layer	Overall Accuracy	Overall Precision	Overall Recall	F1 score
LSTM	2-4	1	29.7%	24.13%	29.68%	0.2662
LSTM	2-4	2	98.6%	98.73%	98.67%	0.9870
LSTM	2-4	3	100%	100%	100%	1.0000
Proposed LSTM	2-4	1	99.93%	99.92%	99.96%	0.9994
Proposed LSTM	2-4	2	99.93%	99.92%	99.96%	0.9994
Proposed LSTM	2-4	3	99.93%	99.94%	99.93%	0.9993

Table 8. Performance summary of the proposed bidirectional LSTM in NSRDB analysis 2 of Figure 30 (b).

Type of Cell/Unit	Input Sequence Length (in Number of Beats)	Number of Hidden Layer	Overall Accuracy	Overall Precision	Overall Recall	F1 score
LSTM	2-4	1	99.96	99.96	99.96	0.9996
LSTM	2-4	2	100	100	100	1.0000
LSTM	2-4	3	5.5	0.31	0.58	1.0000
Proposed LSTM	2-4	1	100	100	100	0.0058
Proposed LSTM	2-4	2	100	100	100	1.0000
Proposed LSTM	2-4	3	100	100	100	1.0000

Table 9. Performance summary of the proposed bidirectional LSTM in MITDB analysis 1 of Figure 31 (a).

Type of Cell/Unit	Input Sequence Length (in Number of Beats)	Number of Hidden Layer	Overall Accuracy	Overall Precision	Overall Recall	F1 score
LSTM	0-2	1	6.28%	7.40%	6.21%	0.0676
LSTM	0-2	2	38.80%	35.66%	38.83%	0.3717
LSTM	0-2	3	1.87%	0.06%	0.18%	0.0013
Proposed LSTM	0-2	1	81.70%	82.83%	81.68%	0.9780
Proposed LSTM	0-2	2	97.78%	97.77%	97.77%	0.9780
Proposed LSTM	0-2	3	98.53%	98.53%	98.53%	0.9855

Table 10. Performance summary of the proposed bidirectional LSTM in MITDB analysis 2 of Figure 31 (b).

Type of Cell/Unit	Input Sequence Length (in Number of Beats)	Number of Hidden Layer	Overall Accuracy	Overall Precision	Overall Recall	F1 score
LSTM	0-2	1	99.70%	97.92%	97.90%	0.9791
LSTM	0-2	2	99.00%	99.01%	99.00%	0.9900
LSTM	0-2	3	2.21%	0.04%	2.13%	0.0008
Proposed LSTM	0-2	1	98.04%	98.07%	98.04%	0.9806
Proposed LSTM	0-2	2	99.26%	99.28%	99.26%	0.9927
Proposed LSTM	0-2	3	99.73%	99.73%	99.73%	0.9973

Table 11. Performance summary of the proposed bidirectional LSTM in MITDB analysis 3 of Figure 32.

Type of Cell/Unit	Input Sequence Length (in Number of Beats)	Number of Hidden Layer	Overall Accuracy	Overall Precision	Overall Recall	F1 score
LSTM	3	1	98.65%	98.76%	98.85%	0.9981
LSTM	3	2	98.17%	98.42%	98.56%	0.9849
LSTM	3	3	98.55%	98.66%	98.86%	0.9876
LSTM	6	1	97.00%	97.37%	97.49%	0.9743
LSTM	6	2	96.85%	97.21%	97.61%	0.9741
LSTM	6	3	97.92%	98.16%	98.44%	0.9830
LSTM	9	1	97.50%	97.70%	98.07%	0.9788
LSTM	9	2	96.50%	96.69%	97.11%	0.9690
LSTM	9	3	96.49%	96.80%	97.22%	0.9701
Proposed LSTM	3	1	97.79%	98.10%	98.22%	0.9816
Proposed LSTM	3	2	99.37%	99.47%	99.52%	0.9949
Proposed LSTM	3	3	99.20%	99.30%	99.42%	0.9936
Proposed LSTM	6	1	98.71%	98.95%	99.06%	0.9901
Proposed LSTM	6	2	99.57%	99.68%	99.59%	0.9963
Proposed LSTM	6	3	99.71%	99.78%	99.72%	0.9975
Proposed LSTM	9	1	63.80%	67.31%	63.49%	0.6534
Proposed LSTM	9	2	99.10%	99.12%	99.31%	0.9921
Proposed LSTM	9	3	99.80%	99.82%	99.83%	0.9982

Table 12. Performance comparison with state-of-the-art models.

Methods	Dataset	Input Sequence Length (Number of Beats)	Overall Accuracy (%)
Proposed model (classification)	MITDB	3	99.73
		9	99.80
H. M. Lynn et al. [29] (classification)	MITDB	3	97.60
		9	98.40
R. Salloum et al. [40] (classification)	MITDB	3	98.20
		9	100
Q.Zhang et al. [33] (identification)	MITDB	1	91.10
X.Zhang [66] (identification)	MITDB	8	97.80
		12	98.90
Ö. Yildirim [67] (classification)	MITDB	5	99.39

4.2 Discussion

The performance results confirm that the ECG classification satisfies the LSTM and bidirectional LSTM in the NSRDB. Particularly, in the proposed model, the learning corresponded well and showed better classification results than that of the conventional LSTM model in MITDB. Tables 7 and 8 list the performance summary for the NSRDB dataset; Tables 9-11 list the performance summary for the MITDB dataset. Table 12 shows that the proposed model outperforms other state-of-the-art methods by obtaining 99.8% classification accuracy. Although it may seem that the proposed model does not perform better than the model proposed in [40], the proposed model of [40] uses longer input sequences. However, similar to the proposed model, when a short input sequence is used, the performance decreases to 98.2%, whilst the proposed model achieves 99.73%. Therefore, the proposed methodology yields enhanced performance, particularly with short sequences.

The primary reasons for the good performance of the proposed models for ECG classification are as follows: (1) sufficient number of deep layers enabled the model to extract personal features (2) the bidirectional model controlled the sequential and time dependencies within the personal ECG signals (3) the late-fusion technique can simplify the prediction score prior to the softmax layer step.

Ⅴ Conclusion

A novel LSTM-based DRNN architecture for ECG recognition is proposed and experimental evaluation of the proposed model on two datasets is performed. The results confirm that the proposed model outperforms other conventional methods and demonstrates a higher efficiency. This improvement can be attributed to the ability of the model to extract more features of ECG using the deep layers of DRNN. The model can further control the temporal dependencies within the ECG signals. Furthermore, the effect of the input sequence length is evaluated and the relationship between the hidden unit and hidden layer is found. The segmentation and grouping of ECG using the preprocessing procedure can effectively impact a real-time system in the identification and authentication processes. The proposed model performs better with shorter sequences compared with the state-of-the-art methods. This characteristic is useful in real-time personal ECG identification systems that require quick results. This thesis confirms that the proposed bidirectional LSTM-based DRNN is promising for the applications of ECG based real-time biometric identification. The scale of samples in these experiments has lacked, and the results were affected by the hardware environments. In the future, a large-scale experimentation study will be conducted with ordinary human ECG signals: calmness, eating, sleeping, running, walking, etc. Further, the proposed bidirectional LSTM-based DRNN will be extensively evaluated with other ECG signals obtained from individuals of different age groups. The future extensive research studies will aim to prove the robustness and efficiency of the proposed model.

References

[1] S. Osowski, L. T. Hoai and T. Markiewicz, "Support vector machine-based expert system for reliable heartbeat recognition," in IEEE Transactions on Biomedical Engineering, vol. 51, no. 4, pp. 582-589, Apr. 2004.

[2] Philip de Chazal, M. O'Dwyer and R. B. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," in IEEE Transactions on Biomedical Engineering, vol. 51, no. 7, pp. 1196-1206, Jul. 2004.

[3] E. J. da S. Luz, T. M. Nunes, V. H. C. De Albuquerque, J. P. Papa and D. Menotti, "ECG arrhythmia classification based on optimum-path forest", Expert Syst. Appl., vol. 40, no. 9, pp. 3561-3573, Jul. 2013.

[4] O. T. Inan, L. Giovangrandi and G. T. A. Kovacs, "Robust Neural-Network-Based Classification of Premature Ventricular Contractions Using Wavelet Transform and Timing Interval Features," in IEEE Transactions on Biomedical Engineering, vol. 53, no. 12, pp. 2507-2515, Dec. 2006.

[5] Physionet ECG Database. Available online: www.physionet.org (accessed on 12 January 2020).

[6] GalenS. Wagner, Marriott's Practical Electrocardigoraphy 10th edition, LippinmottWilliams&Wilkins, 2000.

[7] 허봉열. "심전도 (EKG)", Journal of the Korean Academy of Family Medicine vol. 2, pp. 14-37, Oct. 1981.

[8] T. W. Shen, W. J. Tompkins and Y. H. Hu, "One-lead ECG for identity verification," Proceedings of the Second Joint 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society Engineering in Medicine and Biology, vol. 1, pp. 62-63, Oct. 2002.

[9] E. Braunwald, Atlas of Heart Diseases Volume IX-Arrhythmias (Electrophysiologic principles, Mosby, Philadelphia, 1996).

[10] ECG PEDIA.ORG, https://en.ecgpedia.org/

[11] Einthoven, W. Über den Einfluß des Leitungswidertstandes auf die Geschwindi gkeit der Quecksilberbewegung im Lippmann'schen Capillarelectrometer und: Überdie Form des menschlichen Ecg, Phlüger's Arch. ges. Physiol, 60, pp. 91-101, 1895.

[12] Chr. Zywietz. "A Brief History of Electrocardiography - Progress through Technology," Biosigna Institure for Biosignal Processing and Systems Research, Hannover, Germany. 2003.

[13] 이성택, 김경섭, "심전도 생체 신호 계측", 전기의세계 vol. 59(7), pp. 14-17, 2010.

[14] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", pp. 148 - 156, Addison-Wesley, 1992.

[15] A. M. Martinez and A. C. Kak, "PCA versus LDA," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 2, pp. 228-233, Feb. 2001.

[16] 홍은혜, 고병철, 변혜란, "PCA와 LDA를 이용한 실시간 얼굴 검출," 한국정보과학회 학술대회논문집, Vol. 29, No.2, p.538-540, Feb. 2002.

[17] T. Cover and P. Hart, "Nearest neighbor pattern classification," in IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21-27, Jan. 1967.

[18] Buduma, Nikhil, and Nicholas Locascio. "Fundamentals of deep learning: Designing next-generation machine intelligence algorithms," O'Reilly Media, Inc., 2017.

[19] I. Odinaka, P. Lai, A. D. Kaplan, J. A. O'Sullivan, E. J. Sirevaag and J. W. Rohrbaugh, "ECG Biometric Recognition: A Comparative Analysis," in IEEE

Transactions on Information Forensics and Security, vol. 7, no. 6, pp. 1812-1824, Dec. 2012.

[20] RN. Kandala , R. Dhuli, P. Pławiak, GR. Naik, H. Moeinzadeh, GD. Gargiulo, and S. Gunnam, "Towards real-time heartbeat classification: evaluation of nonlinear morphological features and voting method", Sensors, vol. 19, no. 23, pp. 5079, Nov. 2019.

[21] A. D. C. Chan, M. M. Hamdy, A. Badre and V. Badee, "Wavelet Distance Measure for Person Identification Using Electrocardiograms," in IEEE Transactions on Instrumentation and Measurement, vol. 57, no. 2, pp. 248-253, Feb. 2008.

[22] K. N. Plataniotis, D. Hatzinakos and J. K. M. Lee, "ECG Biometric Recognition Without Fiducial Detection," 2006 Biometrics Symposium: Special Session on Research at the Biometric Consortium Conference, pp. 1-6, Aug. 2006.

[23] T. Tuncer, S. Dogan, P. Pławiak and U. R.Acharya, "Automated arrhythmia detection using novel hexadecimal local pattern and multilevel wavelet transform with ECG signals," Knowl.-Based Syst., vol. 186, Dec. 2019.

[24] Handbook of Biometrics, Jain A., Flynn P., Ross A.A. (Eds.), Springer, 2008.

[25] A. Fratini, M. Sansone, P. Bifulco and M. Cesarelli, "Individual identification via electrocardiogram analysis", Biomed. Eng. OnLine, vol. 14, no. 1, pp. 1-23, Aug. 2015.

[26] M. Tantawi, K. Revett, A.-B. Salem and M. F. Tolba, "ECG based biometric recognition using wavelets and RBF neural network", Proc. 7th Eur. Comput. Conf. (ECC), pp. 100-105, Jun. 2013.

[27] M. Zubair, J. Kim and C. Yoon, "An Automated ECG Beat Classification System Using Convolutional Neural Networks," 2016 6th International

Conference on IT Convergence and Security (ICITCS), pp. 1-5, Sep. 2016.

[28] B. Pourbabaee, M. J. Roshtkhari and K. Khorasani, "Deep Convolutional Neural Networks and Learning ECG Features for Screening Paroxysmal Atrial Fibrillation Patients," in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 48, no. 12, pp. 2095-2104, Dec. 2018.

[29] H. M. Lynn, S. B. Pan and P. Kim, "A Deep Bidirectional GRU Network Model for Biometric Electrocardiogram Classification Based on Recurrent Neural Networks," in IEEE Access, vol. 7, pp. 145395-145405, Sep. 2019.

[30] S. Kiranyaz, T. Ince and M. Gabbouj, "Real-Time Patient-Specific ECG Classification by 1-D Convolutional Neural Networks," in IEEE Transactions on Biomedical Engineering, vol. 63, no. 3, pp. 664-675, Mar. 2016.

[31] M. Hammad, P. Päawiak, K. Wang and U. R. Acharya, "ResNet - attention model for human authentication using ECG signals", Expert Syst., vol. 10, Mar. 2020.

[32] 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, Mar. 2009.

[33] Q. Zhang, D. Zhou and X. Zeng, "HeartID: A Multiresolution Convolutional Neural Network for ECG-Based Biometric Human Identification in Smart Health Applications," in IEEE Access, vol. 5, pp. 11805-11816, May 2017.

[34] J. Jasche, F.S. Kitaura, T.A. Ensslin, “Digital Signal Processing in Cosmology”, Digital Signal Process, pp. 320 - 329, Jan. 2009.

[35] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. Ward, "Deep Sentence Embedding Using Long Short-Term Memory Networks: Analysis and Application to Information Retrieval," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 24, no. 4, pp.

694-707, Apr. 2016.

[36] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder‐decoder for statistical machine translation," Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP), pp. 1724-1734, 2014.

[37] A. Graves, A. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645-6649, May 2013.

[38] A. Graves and J. Schmidhuber, "Offline Handwriting Recognition with Multidimensional Recurrent Neural Networks," In Proceedings of the Advances in Neural Information Processing Systems, pp. 545‐552, Dec. 2009.

[39] W. Zaremba, I. Sutskever and O. Vinyals, Recurrent neural network regularization, Sep. 2014, [online] Available: https://arxiv.org/abs/1409.2329.

[40] R. Salloum and C. -. J. Kuo, "ECG-based biometrics using recurrent neural networks," 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2062-2066, Mar. 2017.

[41] C. Zhang, G. Wang, J. Zhao, P. Gao, J. Lin and H. Yang, "Patient-specific ECG classification based on recurrent neural networks and clustering technique", Proc. 13th IASTED Int. Conf. Biomed. Eng. (BioMed), pp. 63-67, Feb. 2017.

[42] A. Murad and J.-Y. Pyun, "Deep recurrent neural networks for human activity recognition", Sensors, vol. 17, no. 11, pp. 2556, Nov. 2017.

[43] Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998.

[44] Graves, A. “Supervised Sequence Labelling with Recurrent Neural Networks,” Ph.D. Thesis, Technical University of Munich, Munich, Germany, 2008.

[45] S. Hochreiter, Y. Bengio, P. Frasconi, and J. Schmidhuber. “Gradient flow in recurrent nets: the difficulty of learning long-term dependencies,” In S. C. Kremer and J. F. Kolen, editors, A Field Guide to Dynamical Recurrent Neural Networks. IEEE Press, 2001.

[46] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's neural machine translation system: Bridging the gap between human and machine translation," in arXiv:1609.08144, 2016, [online] Available: http://arxiv.org/abs/1609.08144.

[47] J. Kittler, M. Hatef, R. P. W. Duin and J. Matas, "On combining classifiers," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 3, pp. 226-239, Mar. 1998.

[48] V. Pham, T. Bluche, C. Kermorvant and J. Louradour, "Dropout Improves Recurrent Neural Networks for Handwriting Recognition," 2014 14th International Conference on Frontiers in Handwriting Recognition, pp. 285-290, Sep. 2014.

[49] B.-H. Kim and J.-Y. Pyun, "ECG identification for personal authentication using LSTM-based deep recurrent neural networks", Sensors, vol. 20, no. 11, pp. 3069, May 2020.

[50] M. Yao and B. Vocational, "Research on learning evidence improvement for

KNN based classification algorithm", Int. J. Database Theory Appl., vol. 7, no. 1, pp. 103-110, 2014.

[51] MIT-BIH Database. Available online: www.physionet.org (accessed on 12 Jan. 2020).

[52] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," in IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45-50, Jun. 2001.

[53] A.L. Goldberger, L.A.N. Amaral, L. Glass, J.M. Hausdorff, P.C. Ivanov, R.G. Mark, J.E. Mietus, G.B. Moody, C. Peng, and H.E. Stanley, "Physiobank physiotoolkit and physionet components of a new research resource for complex physiologic signals," Circulation, vol. 101, pp. e215 - e220, Jun. 2000.

[54] 김정균, 이강복, 홍상기, "랜덤 포레스트를 이용한 심전도 기반 생체 인증," 전자공학회논문지, vol. 54-6, pp. 100-105, 2017.

[55] K. Potdar, T. S. and C. D., "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers", International Journal of Computer Applications, vol. 175, no. 4, pp. 7-9, Oct. 2017.

[56] J. Pan and W. J. Tompkins, "A Real-Time QRS Detection Algorithm," in IEEE Transactions on Biomedical Engineering, vol. BME-32, no. 3, pp. 230-236, Mar. 1985.

[57] H. Yoon, S. H. Hwang, J. -W. Choi, Y. J. Lee, D. -U. Jeong and K. S. Park, "Slow-Wave Sleep Estimation for Healthy Subjects and OSA Patients Using R - R Intervals," in IEEE Journal of Biomedical and Health Informatics, vol. 22, no. 1, pp. 119-128, Jan. 2018.

[58] A. Gacek and W. Pedrycz, "ECG signal processing classification and interpretation: a comprehensive framework of computational intelligence," in Springer book, London, 2012.

[59] T. Azeem, M. Vassallo, and N. J. Samani, Rapid Review of ECG Interpretation: Manson Publishing, 2005.

[60] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mane, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viegas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," arXiv 2016, arXiv:1603.04467.

[61] Kingma, DP, Adam, JB, "A method for stochastic optimization," arXiv 2014, arXiv:1412.6980.

[62] M. N. Dar, M. U. Akram, A. Shaukat and M. A. Khan, "ECG Based Biometric Identification for Population with Normal and Cardiac Anomalies Using Hybrid HRV and DWT Features," 2015 5th International Conference on IT Convergence and Security (ICITCS), pp. 1-5, Aug. 2015.

[63] M. N. Dar, M. U. Akram, A. Usman and S. A. Khan, "ECG biometric identification for general population using multiresolution analysis of DWT based features," Proc. 2nd Int. Conf. Inf. Secur. Cyber Forensics (InfoSec), pp. 5-10, Nov. 2015.

[64] C. Ye, M. T. Coimbra and B. V. K. V. Kumar, "Investigation of human identification using two-lead Electrocardiogram (ECG) signals," 2010 Fourth IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 1-8, Sep. 2010.

[65] K. A. Sidek, I. Khalil and H. F. Jelinek, "ECG Biometric with Abnormal Cardiac Conditions in Remote Monitoring System," in IEEE Transactions on

Systems, Man, and Cybernetics: Systems, vol. 44, no. 11, pp. 1498-1509, Nov. 2014.

[66] X. Zhang, Y. Zhang, L. Zhang, H. Wang and J. Tang, "Ballistocardiogram Based Person Identification and Authentication Using Recurrent Neural Networks," 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), pp. 1-5, Oct. 2018.

[67] O. Yildirim, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification," Comput. Biol. Med., vol. 96, pp. 189-202, May 2018.