International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Combining A CNN Auto-Encoder with An MLP To Reduce the Computational Cost of Heart Arrhythmia Detection from ECG

Salam Hussein Khudair[1], Behrouz Tousi[2]

Submitted: 02/05/2024 **Revised**: 15/06/2024 **Accepted**: 22/06/2024

Abstract: Heart disease is one of the leading causes of death worldwide, and cardiac arrhythmia is a major symptom of heart disease. Due to the inherent differences between raw time series data and classical features, it is not possible to take advantage of models such as convolutional neural network (CNN), which have a high ability to analyze ECG data. To overcome this limitation, this paper proposes a new idea for ECG classification. The proposed idea is to use separate inputs for raw data and classical features. Based on the proposed idea, two different models of the deep neural network are presented. A more complex architecture consists of a CNN with a Residual structure and an MLP. The second architecture is proposed by combining a CNN auto-encoder with an MLP to reduce the computational cost and the possibility of implementing the proposed idea in wearable devices that are limited in terms of processing capacity and energy consumption. Also, to approach the imbalance problem of existing ECG databases, new approaches have been proposed for oversample and undersample methods. Experimental results on the standard MIT-BIH Arrhythmia database showed a 14.73% increase in recall criterion by the first architecture and 1.27% by the second architecture, compared to the conventional architecture in the intra-patient paradigm. In-Addition, the proposed oversampling and undersampling methods increased the first architectural recall for the VEB+ class by 6.98% and 2.33%, respectively in the intra-patient paradigm.

Keywords*: ECG signal, CNN, Deep learning, auto-encoder, arrhythmia, MLP.*

1- Introduction

Today, with the improvement in quality of life, cardiovascular diseases have become the leading cause of death in humans. From 2013 to 2016 AD, 48% of individuals over the age of 20 (121.5 million people) in the United States were affected by cardiovascular diseases. Additionally, 31% of global mortality rates in 2016 AD were attributed to these diseases. For this reson various machine learning methods have been proposed for arrythmia detection as we describe here: The softmax function, as discussed in [1], is pivotal in multi-class classification tasks, such as arrhythmia detection, where it's essential to differentiate between various types of heartbeats accurately. Catalani's work [2] exemplifies the application of machine learning techniques in arrhythmia classification, setting a foundation for subsequent research in the domain. Hannun et al.'s study [3] represents a landmark in utilizing deep neural networks for arrhythmia detection, showcasing the potential of these models to achieve cardiologist-level accuracy. Luo et al. [4] further the narrative by proposing a patient-specific deep architectural model for ECG classification, emphasizing the importance of personalized healthcare solutions. The statistical reports by Virani et al. [5] and Benjamin et al. [6] provide a comprehensive overview of heart disease and stroke statistics, underlining the critical need for effective arrhythmia detection mechanisms. In a similar vein, Sannino and De Pietro [7] demonstrate the efficacy of deep learning approaches in ECG-based heartbeat classification, contributing to the growing body of evidence supporting AI in healthcare. Romanò's text atlas [8] and the interpretability analysis by Li et al. [9] offer insights into the practical aspects of ECG interpretation and the challenges of classifying heartbeats using machine learning models. Yao et al. [10] and the systematic review by Hong et al. [11] explore the advancements in arrhythmia detection algorithms, highlighting the role of attention mechanisms and convolutional neural networks in improving classification accuracy. Ebrahimi et al. [13] provide a critical review of deep learning methods for ECG arrhythmia classification, pointing out the challenges and opportunities in the field. The foundational texts by Francois [14], Géron [15], and Hastie et al. [16] are essential resources for understanding the underlying theories of machine learning and deep learning, which are crucial for developing sophisticated arrhythmia detection models. The significance of publicly available databases, such as the MIT-BIH Arrhythmia Database [17] and PhysioNet [18, 25], cannot be overstated, as they enable researchers to train, test, and validate their models on standardized datasets. Llamedo and Martínez [19] discuss the importance of feature selection in heartbeat classification, a theme that is echoed in the work of Halperin and Hart [20], Kannel et al. [21], and

[1,2] *Department of electrical and computer engineering, Urmia University, Urmia, Iran*
Salamhussen11@gmail.com
Corresponding author: ***Behrouz Tousi,*** *b.tousi@urmia.ir*

Wolf et al. [22], who explore the epidemiological aspects of atrial fibrillation and its role as a risk factor for stroke. Recent studies, such as those by Pourbabaee et al. [23], Moody et al. [24], and Li and Boulanger [26], delve into the application of deep convolutional neural networks and data augmentation methods for enhancing the detection of arrhythmias, including atrial fibrillation. Harrigan et al. [27], Das and Ari [28], and Goodfellow et al. [29] contribute to the understanding of signal processing techniques and the use of convolutional neural networks in ECG classification. Xu and Liu [30], Acharya et al. [31], and Gao et al. [32] further demonstrate the potential of deep learning models, including CNNs and LSTMs, in accurately identifying arrhythmic events from ECG signals. The works of Singh and Tiwari [33], Cao et al. [34], and Donoho [35, 36] explore various aspects of signal denoising and augmentation, crucial for preparing ECG data for analysis. Shaker et al. [37] leverage the power of Generative Adversarial Networks (GANs) to generalize Convolutional Neural Networks (CNNs) for ECG classification, showcasing the utility of synthetic data in overcoming the limitations posed by imbalanced datasets. This is complemented by the studies of Thakor et al. [38] and Alfaras et al. [39], who focus on the optimization of signal processing techniques and machine learning models for efficient heartbeat classification, illustrating the ongoing efforts to refine the analytical capabilities of arrhythmia detection systems. The works of Liu et al. [40], Alvarado et al. [41], and Baghdadi [42] delve into the complexities of ECG signal analysis, employing sophisticated models to enhance the precision of arrhythmia detection. These studies highlight the significance of addressing the nuances of ECG signals, such as the variability in signal quality and the presence of noise, which are critical challenges in the development of reliable detection systems. Yildirim's research [43, 44] introduces innovative deep learning models, including bidirectional LSTM network models and deep convolutional neural networks, for the classification of ECG signals. These approaches exemplify the shift towards more advanced neural network architectures that offer improved accuracy and efficiency in arrhythmia detection. Mousavi and Afghah [45] and Jiang et al. [46] present novel neural network systems designed to handle imbalanced datasets, a common issue in medical data analysis. Their work underscores the importance of developing algorithms that are capable of distinguishing between minority classes, such as rare arrhythmias, without being overwhelmed by the majority classes. De Chazal et al. [47] and Luz et al. [48] focus on the automatic classification of heartbeats using a combination of ECG morphology and heartbeat interval features, further advancing the field by integrating multiple data dimensions into the classification process. This multi-feature approach is indicative of the trend towards creating more holistic models that consider various aspects of the ECG signal for a more accurate diagnosis. Mathews et al. [49] and Golpayegani [50] explore the application of deep learning and neural networks in ECG beat classification, demonstrating the versatility of these technologies in extracting meaningful patterns from complex data. Kachuee et al. [51] and Al Rahhal et al. [52] highlight the potential of transfer learning and deep learning approaches, respectively, in improving the classification of electrocardiogram signals, emphasizing the benefits of leveraging pre-trained models and deep architectures for enhancing performance. The contributions of Amirshahi and Hashemi [53], Saadatnejad et al. [54], and Hadaeghi [55] exemplify the integration of advanced computational models into wearable devices for continuous monitoring, pointing towards the future of personalized healthcare and the potential for real-time arrhythmia detection. Data augmentation techniques, as explored by Hatamian et al. [56] and the synthetic minority over-sampling techniques discussed by Chawla et al. [57] and He et al. [58], address the critical issue of class imbalance in training datasets. These methodologies enable the development of models that are more robust and generalizable, thereby improving the accuracy of arrhythmia detection across diverse patient populations. Generative models, such as those described by Goodfellow et al. [59], Delaney et al. [60], and Radford et al. [61], offer innovative solutions for generating synthetic ECG signals that can enhance the training of machine learning models. These approaches underscore the creative use of AI to overcome data limitations, paving the way for more sophisticated and capable arrhythmia detection systems. The ongoing research and development in heart arrhythmia detection underscore a pivotal shift towards harnessing computational intelligence to address critical healthcare challenges.

2- Problem statement

In this section, the proposed problem is described. In Section 1, the data preparation, extraction of classical features, and the construction of training and evaluation sets are explained. Section 2 introduces two new methods to address the imbalance in the training set. In Sections 3 and 4, our proposed deep learning models for heartbeat classification, using deep learning alongside classical features, are presented. Section 5 explains how to determine hyperparameters and train the proposed models.

2-1- Data Preparation

In this paper, similar to the majority of work conducted in this field, the MIT-BIH database is utilized as the database. Furthermore, according to the ANSI/AAMI EC57:1998 recommendation, 4 records containing paced beats, most of which belong to class Q, are excluded (other records do not contain Q beats), and classification into 4 classes of AAMI (N, V, S, and F) is performed. Before dividing this database into training and evaluation sets, several preprocessing steps have been applied to the available records (Fig. 3-1). These preprocessing steps include noise removal from the records, segmentation into independent beats, and normalization.

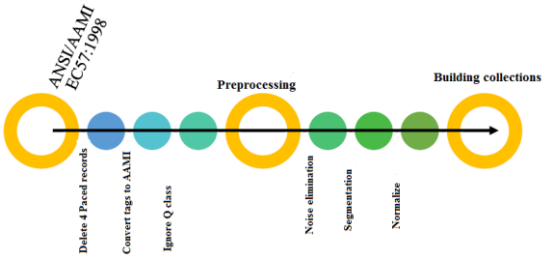

Fig. 3-1: Preprocessing steps performed on data prior to dividing them into training and evaluation sets.

3-1-1- ECG Preprocessing and Beat Segmentation

Initially, according to the proposed method in [38] and [37], a second-order Butterworth filter with a frequency range of [0.5-45Hz] is used to eliminate low and high-frequency noise. The Butterworth filter is designed to have a flat frequency response within the passband as much as possible. This filter has less computational complexity compared to wavelet-based methods. Fig. 3-2 illustrates the original ECG signal (before filtering) and the filtered signal along with the frequency response of the Butterworth filter with the mentioned characteristics.

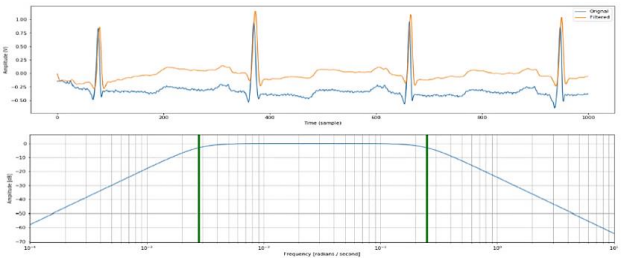

Fig. 3-2: (a) ECG signal before and after filtering, (b) frequency response of the Butterworth filter.

After noise removal from the records, RRI features are extracted for each heartbeat. These features are as follows:

- Pre RRI: The distance between the R-peak of the current beat and the R-peak of the previous beat.
- Post RRI: The distance between the R-peak of the current beat and the R-peak of the next beat.
- Local RRI: The local average RR-Interval for the current beat.
- Global RRI: The average of RR-Intervals for all beats in a record.

These features are calculated using the following equations: 3-1, 3-2, 3-3, and 3-4.

$$\alpha_n = \theta_n - \theta_{n-1}$$

$$\beta_n = \theta_{n+1} - \theta_n$$

$$l\gamma_n = \frac{\sum_{i=-j}^{k} \gamma_{n+i}}{j+k+1}$$

$$g\gamma_\Psi = \frac{\sum_{n\in\Psi} \gamma_n}{|\Psi|}$$

In these relationships, n refers to the beat number, $\theta_n$represents the R-Peak location, $\alpha_n$denotes the Pre RRI feature, $\beta_n$indicates the Post RRI feature, and $l\gamma_n$represents the Local RRI feature of the nth heartbeat. Additionally, i and j refer to the number of left and right neighbors in the Local RRI feature, while γ one of the Local RRI or Global RRI features is represented by $g\gamma_\Psi$ in the Ψ record. $|\Psi|$denotes the number of beats present in the Ψ record. These features are utilized further for segmentation and classification purposes. In this context, considering the objective of the problem, similar to most conducted tasks, the R-Peak locations determined by the database itself are employed. After noise removal and feature extraction of the RRI records, according to the suggested method [37], the records are divided into independent beats. In this method, the range of each beat is determined by placing its R-Peak position within the $[\frac{-1}{3}PreRRI\ .\frac{2}{3}\ PostRRI]$ interval (Equation 3-5). This segmentation method, in addition to being dynamic and accurate, unlike methods that require detection of P and T wave locations, does not add much processing load to the system. Especially considering that Pre-RRI and Post-RRI features are also used during classification. The results of applying this method are shown in Fig. 3-3.

$$X_{\mathrm{N}} = \mathbf{\Psi}_{\left[\theta_n - \frac{1}{3}\alpha_n . \theta_n + \frac{2}{3}\beta_n\right]}$$

After the beat segmentation, the data of each beat are normalized between 0 and 1 using Equation 3-6. In this equation, x_t and $\bar{x}_t$ are the original and normalized samples at moment t within the beat (X).

$$\bar{x}_t = \frac{x_t - \min(X)}{max(X) - \min(X)}$$

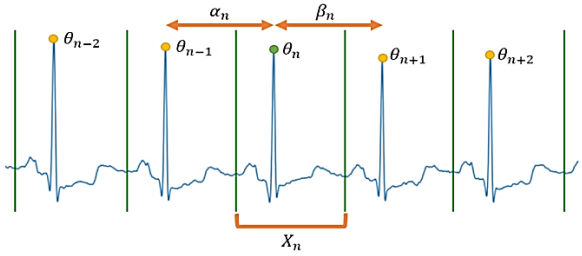

Fig. 3-3: Segmentation of ECG record into individual beats.

In the context of the input length of the model needing to be constant, the length of all pulses is set to 300 samples. If the length of a pulse is less than 300 samples, same padding is applied at the beginning and end of the pulse to increase its length. Conversely, if the length of a pulse exceeds 300 samples, samples are removed from the beginning and end of the pulse to decrease its length.

3- Proposed method

3- 3- Proposed Model One: CNN - Hybrid with Residual Structure

As mentioned in Section 2, deep learning models have shown higher efficiency compared to classical feature extraction methods. However, several decades of research have been devoted to classical features, and some of them have proven to be important. For example, RRI features are not only used in automated systems but also by physicians in disease diagnosis. For this reason, in some studies, a combination of these features with deep learning has been employed, which was discussed in Section 2.6.3.

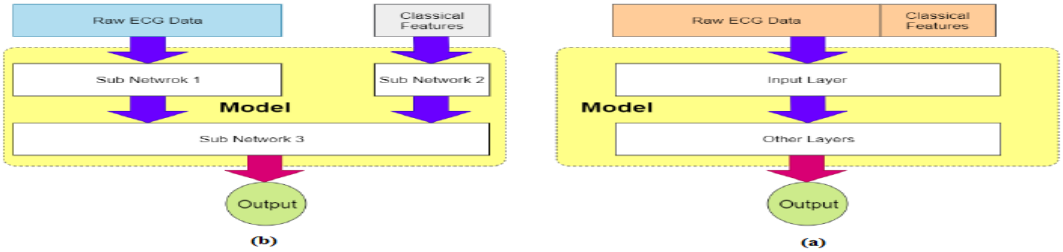

Fig. 3-8: (a) Previous model architectures and (b) Proposed model architecture for incorporating classical features alongside deep learning.

The architectural and proposed methods for combined utilization of classical features and deep learning are shown in Fig. 3-8, this architecture enables the design of two independent sub-networks, each suitable for its own type of input. The general equations of conventional models and the proposed model are given in equations 3-8 and 3-9, respectively. In these equations, P represents the prediction vector (output of the model), F is the model function, X is the raw pulse data vector, R is the vector of classical features, and the operation of concatenating two vectors is denoted by ⧺. Essentially, the previous models are single-variable functions (vector-type) and therefore behave uniformly with all elements of the input vector. However, the proposed model is a two-variable function, allowing it to interact differently and independently with the inputs. The proposed model is illustrated with more detail in Fig. 3-9 and will be discussed further in the subsequent section.

$$P = F(X \shortparallel R)$$

$$P = F(X.R)$$

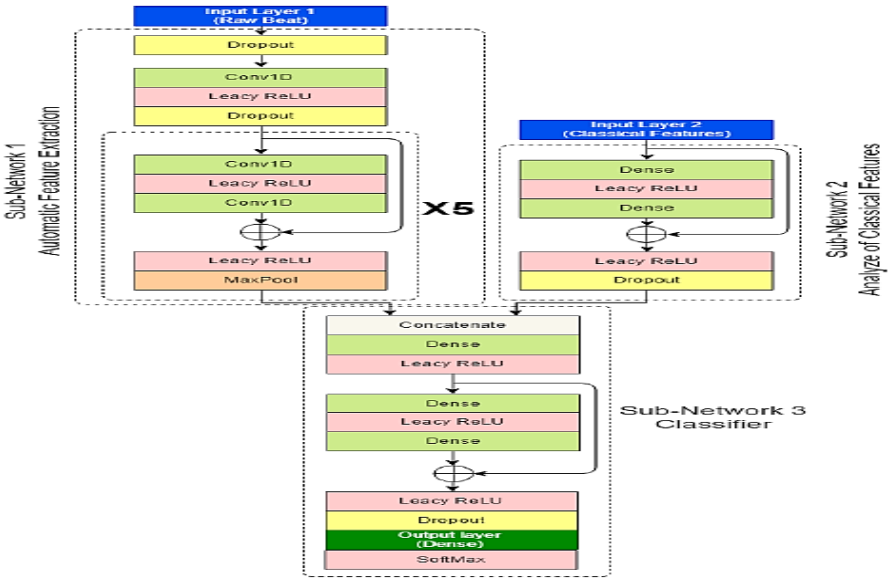

Fig. 3-9: Proposed composite CNN model.

3-3-1- Sub-network 1: Automatic Feature Extraction Sub-network

Subnetwork 1 is responsible for automatically extracting features from raw input data (e.g., heartbeat) using deep learning. Considering the success of CNN in previous research, it has been used in this study as well. The proposed CNN is inspired by the proposed networks (51) and (3) and consists of one standard convolutional block and five residual convolutional blocks. The structure and advantages of residual blocks are explained in Section 2. To prevent overfitting of the model on input data, especially when the model is trained on a balanced dataset augmented by data augmentation, a Dropout layer has been used after the input and the first convolutional block. During model training, the Dropout layer randomly removes some neurons from the input layer (setting their values to zero), thus introducing changes in the model's structure. Each residual convolutional block utilizes two serial convolutional layers, a skip connection, and a MaxPool layer for spatial feature reduction. The number of filters in the convolutional layers in the first (standard convolution), second, and third (residual convolution) blocks is 32, while in the fourth and fifth blocks, it is 64, and in the last block, it is 128. Considering the halving of feature spatial dimensions after each MaxPool layer and the input heartbeat length (300 samples), the output dimensions of the first subnetwork are 9x128 (1152 features). The change in feature space from the input layer to the last subnetwork 1 block is illustrated in Fig. 3-10. It is expected that this network, due to its high number of layers (11 convolutional layers) and the residual structure, learns high-level features from input heartbeats.

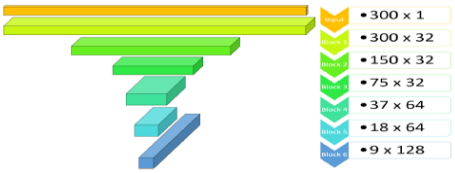

Fig. 3-10: Dimensions alteration of the feature space after each block in the subnetwork for automatic feature extraction (subnetwork 1).

3-3-2- Subnetwork for classical feature analysis (Subnetwork 2)

This sub-network is responsible for analyzing the classic features of the input. The network consists of 2 dense layers along with a skip connection. The number of neurons in the two dense layers is 16 and 8 respectively. The skip connection allows the input features of this sub-network to be directly deposited into sub-network 3. At the end of this sub-network, a Dropout layer is placed. The input layer of this model consists of 5 neurons, which will be explained further. Four RRI features described in the previous section, along with the duration of the heartbeat (length of the heartbeat before it reaches 300 samples), are considered as the input feature vector of this sub-network (Fig. 3-11). The reason for selecting these features as input to the proposed model is that these features become independent during signal segmentation into individual heartbeats, and although the model is strong, it has no chance of learning them.

Pre-RRI	Post-RRI	Local-RRI	Global-RRI	Beat duration

Fig. 3-11: Input Feature Vector below Network 2

3-3-3 subnetwork classifier (Subnetwork 3)

This subnetwork plays the role of a classifier model. The input to this network is two output vectors from previous

subnetworks, which are concatenated at the beginning of this subnetwork. This network consists of three dense layers along with a skip connection and a dropout layer. The output layer is also a dense layer with 4 neurons and the SoftMax activation function. For all convolutional and dense layers in the proposed model, unlike most previous models that use the ReLU activation function, the proposed method uses the Leaky ReLU function. The derivative of the ReLU function is 0 when its input is less than 0, which causes some neurons to always remain in the negative value without any change. These neurons are commonly referred to as "dead neurons." One of the methods to prevent this phenomenon is to use the Leaky ReLU function (Equation 3-10). This function also has a non-zero derivative (a) for inputs smaller than zero, allowing the neurons to learn even with negative values. In Equation 3-10, a is a predefined constant value. Fig. 3-12 illustrates the Leaky ReLU function.

$$LeakyReLU(x) = \begin{cases} x. & x > 0 \\ ax. & x \leq 0 \end{cases} \quad (10\text{-}3)$$

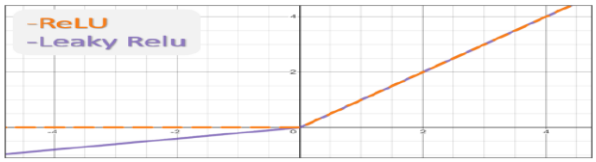

Fig. 3-12: ReLU and Leaky ReLU Functions ($\alpha = 0.2$)

3-4- Proposed Model - Combined Auto-Encoder

In the previous section, a combined CNN with a residual structure was proposed using skip connections, resulting in a model with a large number of layers. However, increasing the number of layers and the use of skip connections impose computational and memory requirements, posing challenges for implementing low-power and small wearable devices. Therefore, in this part of the research, a simpler combined model without the residual structure is proposed. The proposed model consists of an auto-encoder for feature extraction from the pulse signals and a MLP for classifying the pulse type based on the extracted features from the encoder and the classical features used in the previous section. As explained in Section 2, the methods that have been used so far for classification have initially trained the auto-encoder independently and then used its encoder as a feature extraction model (Fig. 3-13). In this case, considering that the main goal of the auto-encoder is to reconstruct the input signal as accurately as possible, it is possible for the encoder to learn features that are important for reconstruction but not suitable for classification, or to not learn features that are important for classification but not useful for reconstruction. To investigate this issue, we trained two auto-encoders, one composed of dense layers and the other composed of convolutional layers, only on one class of pulses (using the training dataset) and calculated their reconstruction errors on all four classes in the evaluation set. To calculate the reconstruction error, we used the mean absolute error (MAE) function according to Equation 3-11. In this equation, E_{AE} is the reconstruction error of the auto-encoder, X is the input pulse, $\bar{X}$ is the reconstructed pulse, and $|X|$ is the length of the pulse. Histograms of reconstruction errors for the existing samples in the evaluation dataset are shown in Fig. 3-14 for the auto-encoder with dense layers and in Fig. 3-15 for the auto-encoder with convolution layers, for four types of beats: N, V, S, and F. Although the models have only been trained on the normal class samples in the training set, it is expected that the reconstruction errors for other classes, as they have different characteristics, would be high. However, based on the presented histograms in Fig. 3-14 and Fig. 3-15, the model's errors are approximately within the same range for all four classes. This may be due to the learned features by the encoder being specific to reconstructing heartbeat signals and not suitable for proper classification. Fig. 3-16 illustrates several reconstructed heartbeat samples by the convolutional auto-encoder trained on class N. As observed in this Fig., the auto-encoder is successful in reconstructing samples from all classes.

$$E_{AE} = \frac{\sum |X - \bar{X}|}{|X|} \quad (10\text{-}3)$$

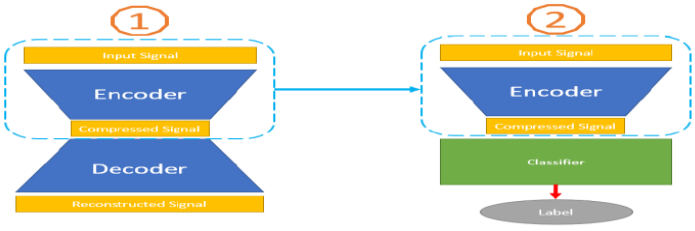

Fig. 3-13: Common Usage of Auto-Encoder for Classification.

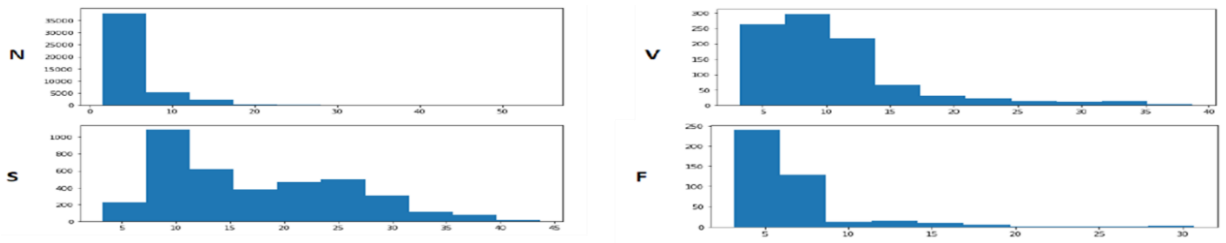

Fig. 3-14: Histogram of reconstruction errors for N, V, S, and F beats in a compact trained auto-encoder applied to the N class.

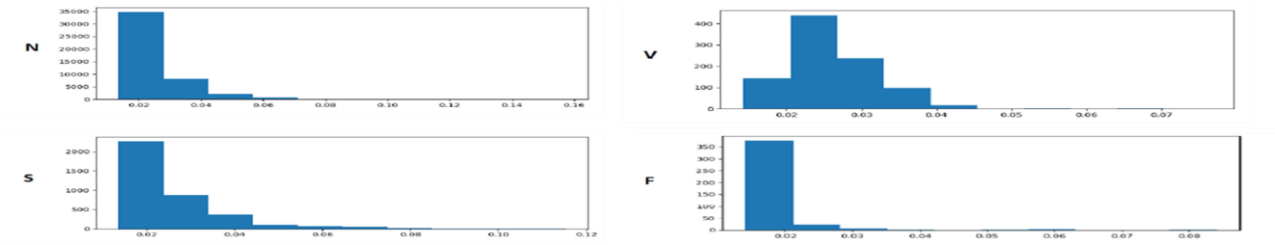

Fig. 3-15: Histogram of reconstruction errors for N, V, S, and F beats of the auto-encoder trained on class N.

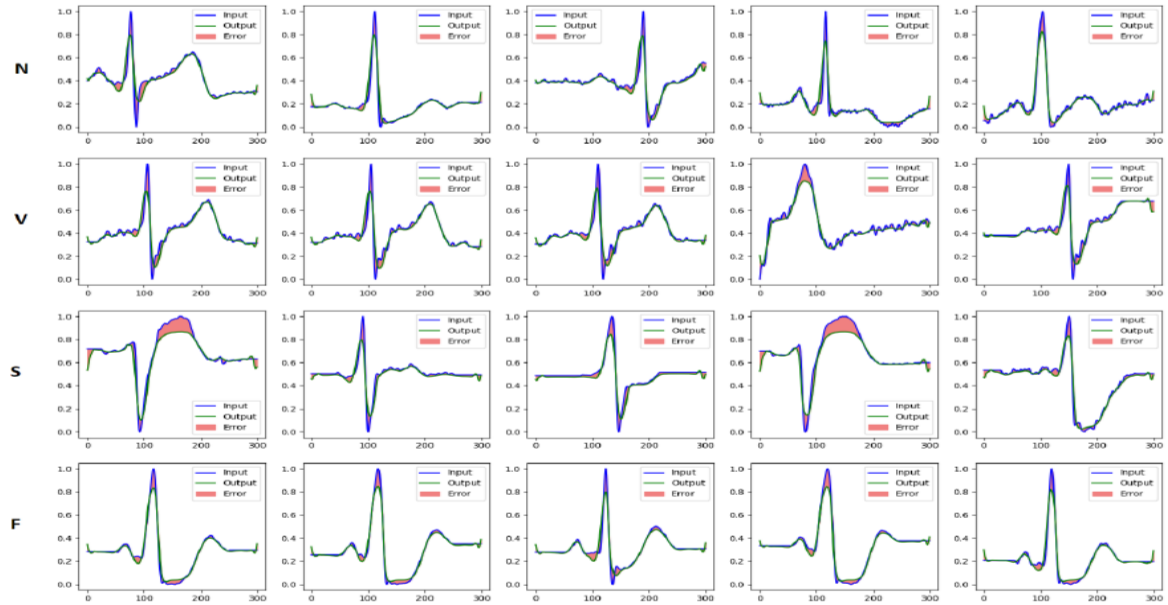

Fig. 3-16: Sample reconstructed beats generated by Auto-Encoder Convolution trained on class N.

To ensure that the encoder pays attention to important features for beat classification during training, our proposed Auto-Encoder combines two separate input vectors and two separate output vectors. The encoder's output is connected to both the decoder for beat reconstruction and the MLP for beat type determination. The overall structure of the proposed Auto-Encoder is shown in Fig. 3-17. In this architecture, the model learns in an integrated manner, and therefore, from the beginning of the training process, the loss value in beat classification affects the weights of the connections under the encoder network (through backpropagation). Considering that the goal of designing this model is its implementation in wearable devices, we have tried to avoid using layers and structures that would complicate the model or have a large number of layers. The proposed model's encoder consists of three convolutional blocks. To reduce the dimensionality of the input vector, MaxPool layers with a stride of 2 are used in the first two blocks. As a result, the output length of the encoder is one-fourth of the input pulse length. For the proposed decoder model, we have used four convolutional blocks. In the first two blocks, two UpSample layers are included to increase the vector length and bring it to the original pulse length. Each UpSample layer repeats each element of its input vector once. Therefore, the output vector length is twice its input (Fig. 3-18). For the classifier subnetwork in the proposed model, we have used an MLP with four dense layers (including the output layer). The details of the proposed hybrid auto-encoder are shown in Fig. 3-19.

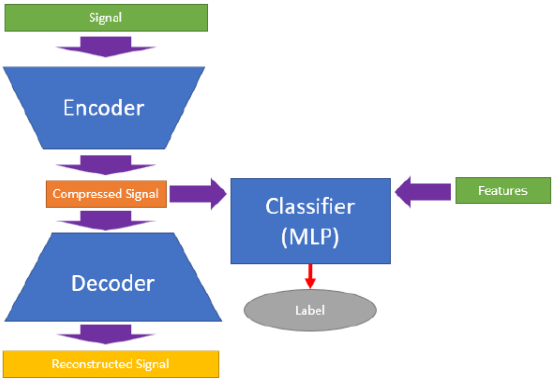

Fig. 3-17: Proposed Hybrid Auto-Encoder Structure.

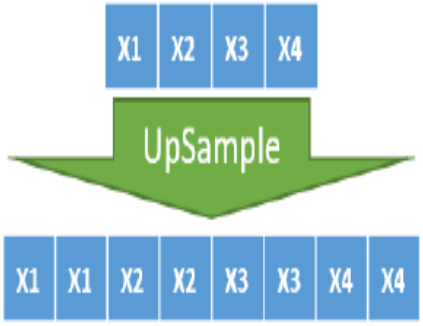

Fig. 3-18: UpSample Layer

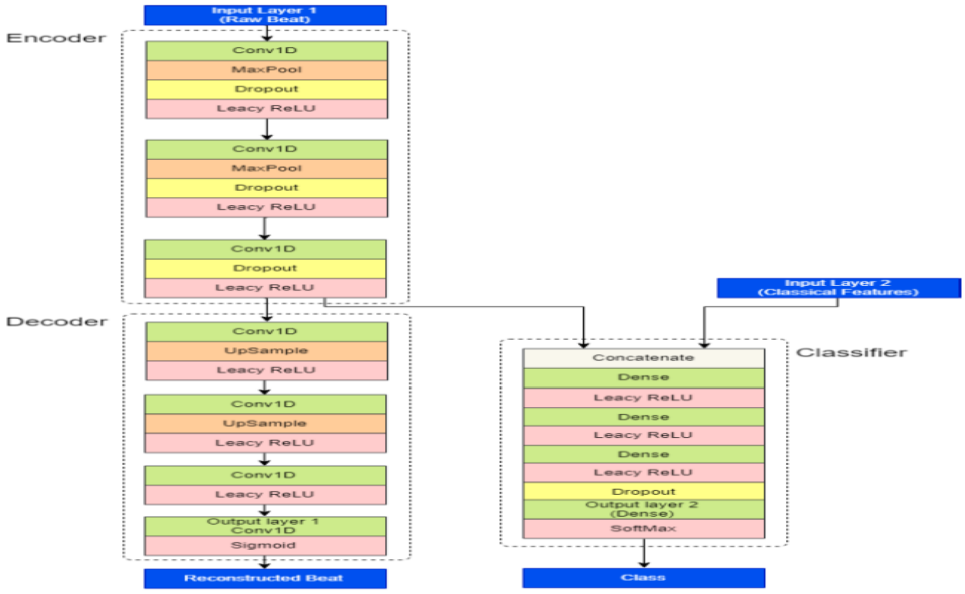

Fig. 3-19: Proposed Auto-Encoder Integration

3 -5- Determining Hyper-parameters and Model Training

To determine the optimal hyper-parameters in the proposed model, we have employed a grid search technique. In grid search, all possible combinations in the search space are examined. The searched meta-parameters include the learning rate, number of training epochs, batch size, and the dropout rates of the layers. The grid search operation to find suitable hyper-parameters for the model is carried out independently for all three patterns. The network training is performed using the Adam optimizer, and the Cross-Entropy function is used to calculate the loss. Here, we have not utilized any technique for early stopping of the training process.

4- Results and Analysis

4-1- Tools used for implementation and evaluation of proposed methods

In this paper, the Python programming language (version 3.7) based on the Windows 10 operating system has been used to write all programs. The hardware platform used is a PC with the following specifications: Intel Core i5 processor, 8 gigabytes of memory, and NVIDIA GeForce GTX960 - 4GB graphics processor.

4-2- Metrics used in evaluating the performance of proposed methods:

When evaluating classifier models, the class assigned by the dataset itself to the data, known as the true class, and the class predicted by the model for each input data, known as the predicted class, are considered. In binary classifiers (two classes: positive (P) and negative (N)), TP (True Positive) and TN (True Negative) represent the number of positive and negative samples correctly classified, respectively. FP (False Positive) represents the number of negative samples incorrectly classified as positive, and FN (False Negative) represents the number of positive samples incorrectly classified as negative. These numbers are used to measure the performance of classifier models.

4-3- Investigation of constructed sets

For the pre-processing and extraction of classical features, two features, Local RRI and Global RRI, have been calculated based on the Pre RRI feature (in Eq. 3-3 and Eq.3-4, $\gamma = a$ is considered). Additionally, a

neighborhood in Local RRI has been defined as 10 adjacent beats (in Eq. 3-3, $= 5. i = 5$). Summary of the number of samples in each data set in three patterns: Inter-Patient, Intra-Patient and Patient-Specific is given in Table 3-4. As mentioned before, in the Inter-Patient pattern, 22 records were used for the training set (DS1 in Table 2-1) and 22 other records (DS2 in Table 2-1) were used for the evaluation set. In the Intra-Patient pattern, 50% of the samples from each beat were randomly selected for the training set, and the remaining 50% were used for the evaluation set. In the Patient-Specific pattern, 22 records from DS1 were used for training the general models, and the initial 20% of each record in DS2 were used for training the specific models (22 specific models), while the remaining 80% of each record were used for evaluating the specific models. Based on Table 4-3, approximately 90% of all beats in the datasets belong to the normal class.

Table 4-3: Summary of the constructed training and evaluation datasets in three patterns: Inter-Patient, Intra-Patient, and Patient-Specific.

Evaluation collection / Dedicated training-evaluation				**Education collection / general education**				**Pattern**	**Row**
F	**S**	**V**	**N**	**F**	**S**	**V**	**N**		
388	3221	1833	44235	415	3788	943	45845	Inter-Patient	**1**
401	3504	1388	45040	402	3505	1388	45040	Intra-Patient	**2**
388	3221	1833	4435	415	3788	943	45845	Patient-Specific	**3**

4 -4- Evaluation of the First Model: Hybrid CNN with Residual Structure

In this section, a proposed Hybrid CNN model is evaluated using three patterns: Inter-Patient, Intra-Patient, and Patient-Specific. Subsequently, the obtained results for each pattern are presented and discussed.

4-4-1-Model Evaluation in Inter-Patient Pattern

In the Inter-Patient pattern, the proposed model was evaluated on the reference (imbalanced) training set and the balanced training set achieved by increasing the proposed samples using SMOTE, SVMSMOTE, ADASYN, and reducing the conventional and proposed training samples. Additionally, an experiment was conducted on the robustness of the model against the reference set imbalance by assigning weights to the existing classes (Row 2). After training the model in each of these scenarios, the performance of the model was evaluated on the evaluation set without any manipulation or changes. The confusion matrix of the three reference experiments, the proposed sample increase, and the proposed sample reduction are presented in Table 4-4. The following conclusions can be drawn from this table:

1- According to Table 4-4 (a), in the case where the model has been trained on a reference training set where the number of samples N is much larger than other classes, it has made a significant number of false normal predictions for V, S, and F beats, indicating a bias towards these classes. Therefore, model retrieval in this scenario is low for the V, S, and F classes (Row 1, Table 4-5).

2- When the model is trained on the balanced dataset by increasing the proposed samples, it prevents the model from biasing towards class N. As a result, the number of FN cases for classes V and S compared to the previous state decreased significantly from 747 and 328 to 173 and 83, respectively. Following this, the recovery of these two classes increased from the reference state of 0.57 and 0.48 to 0.82 and 0.84, respectively. However, considering the increase in false positive alerts (FP) for normal beats from 251 and 78 to 809 and 127, it can be concluded that the examined data augmentation techniques are not successful in generating new useful information and only prevent model bias.

Table 4-4: Confusion Matrix (a) in the reference training set, (b) after oversampling, and (c) after under-sampling in the Inter-Patient pattern.

(A)	**Predicted**			
	N	**V**	**S**	**F**

Actual					
	N	43787	251	78	122
	V	747	1045	38	6
	S	328	1075	1562	256
	F	112	3	2	271

(B)

Actual \ Predicted	**N**	**V**	**S**	**F**
N	38836	809	127	4466
V	173	1513	136	14
S	83	314	2721	103
F	149	4	7	228

(C)

Actual \ Predicted	**N**	**V**	**S**	**F**
N	36502	823	259	6654
V	250	1448	62	76
S	204	297	2649	71
F	76	3	7	302

Table 4-5: Precision (PRE) and CNN-based Retrieval (REC) of the proposed combined model in the Inter-Patient pattern for four AAMI classes.

Total		**F**		**S**		**V**		**N**			
REC	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**Description**	**Row**
0.69	0.69	0.70	0.41	0.48	0.93	0.57	0.44	0.99	0.97	Reference Dataset	**1**
0.73	0.61	0.58	0.08	0.78	0.82	0.62	0.57	0.93	0.98	Class Weight	**2**
0.59	0.55	0.52	0.02	0.64	0.87	0.44	0.35	0.74	0.97	SMOTE DA	**3**
0.69	0.52	0.49	0.05	0.73	0.84	0.76	0.20	0.8	0.98	SVMSMOTE DA	**4**
0.76	0.54	0.82	0.03	0.67	0.8	0.86	0.33	0.71	0.99	ADASYN DA	**5**
0.62	0.52	0.15	0.01	0.93	0.66	0.56	0.42	0.82	0.99	Traditional Under-sampling	**6**
0.78	**0.63**	**0.59**	**0.05**	**0.84**	**0.91**	**0.83**	**0.57**	**0.88**	**0.99**	**Proposed Oversampling DA**	**7**
0.80	**0.62**	**0.78**	**0.04**	**0.82**	**0.89**	**0.76**	**0.56**	**0.83**	**0.99**	**Proposed Under-sampling**	**8**

The results in Tables 4-5 are presented as bar charts in Fig. 4-1. Based on Tables 4-5 and Fig. 4-1, the following conclusions can be drawn:

1- The experiment demonstrated a decrease in both accuracy and recall measures for class F when increasing the sample size.

2- The techniques for dealing with database imbalance include model regularization, strengthening the training set, and reducing the training set examples (rows 2 to 8 in Table 4-5). However, these approaches significantly reduced the accuracy for class F from 0.41 in the reference condition (row 1 in Table 4-5) to less than 0.1 (almost zero). Considering that class F is closely related to class N in terms of morphology, when preventing the model from being biased towards class N, a greater number of normal beats are mistakenly labeled as class F. For example, in the sample expansion experiment, 4466 normal beats have been misclassified as class F. This number was 122 in the reference experiment.

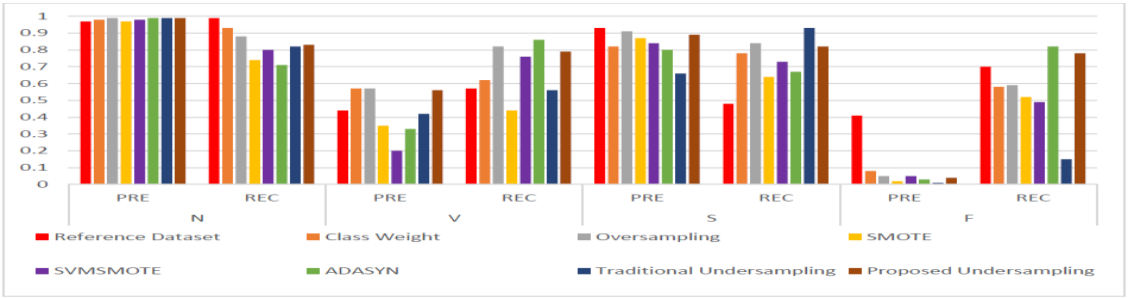

Fig. 4-1: Proposed Hybrid Precision (PRE) and CNN-based Retrieval (REC) in Inter-Patient Pattern for Four AAMI Classes.

The results obtained from these experiments are presented in Table 4-6 for four classes (N, AAMI, V, S, and F) based on two evaluation metrics: F1 and Accuracy (ACC). Based on Table 4-6, the following conclusions can be drawn:

1. According to Row 1 of Table 4-6, when the model is trained on the reference training set, it achieves an overall F1 score of 69.0 and an accuracy of 94.0. Comparing this row with Rows 2 to 6 of Table 4-6, it can be observed that all methods for addressing the imbalanced training set had a negative impact on the overall accuracy and F1 score. The main reason for the decrease in the overall F1 score is the significant reduction in the F1 score for class F, which, as explained earlier, is caused by the decreased precision of class F in these experiments.

2. Rows 7 and 8 of Table 4-6 demonstrate that only in two proposed methods of oversampling and under-sampling in this paper, there is a slight improvement in the overall F1 score. In these experiments, the F1 scores for classes V and S have increased, but similar to previous experiments, the F1 score for class F had a significant decrease, almost neutralizing the improvement in the F1 scores for classes V and S.

3. In general, the only methods that were able to increase the recall for classes V and F, which are medically important, while maintaining the overall F1 score (compared to the reference state), were our two proposed methods of under-sampling and oversampling. The SMOTE method and its derivatives generate new data by using neighborhoods, which means that the generated data may not be accurate. However, our proposed oversampling method generates new data by making small changes to the existing samples of each class, resulting in a lower probability of generating incorrect data. Our proposed under-sampling method only balances the dataset and does not generate any new data.

Table 4-6: F1 and Accuracy of the Hybrid CNN Model in the Four AAMI Classes in the Inter-Patient Pattern.

Improvement(%)		Total		F	S	V	N		
ACC	**F1**	**ACC**	**F1**	**F1**	**F1**	**F1**	**F1**	**Description**	**Row**
-	-	0.94	0.69	0.52	0.64	0.50	0.98	Reference Dataset	**1**
-4.25	-4.34	0.90	0.66	0.14	0.80	0.60	0.95	Class Weight	**2**
-23.40	-17.39	0.64	0.72	0.04	0.74	0.39	0.84	SMOTE DA	**3**
-15.95	-14.49	0.73	0.79	0.08	0.78	0.32	0.88	SVMSMOTE DA	**4**
-24.46	-8.69	0.67	0.71	0.06	0.73	0.48	0.83	ADASYN DA	**5**
-13.82	-18.84	0.93	0.81	0.02	0.78	0.48	0.90	Traditional Under-sampling	**6**
-7.44	**1.44**	**0.84**	**0.87**	**0.09**	**0.88**	**0.68**	**0.93**	**Proposed Oversampling DA**	**7**
-12.76	**1.44**	**0.82**	**0.82**	**0.08**	**0.85**	**0.66**	**0.90**	**Proposed Under-sampling**	**8**

In table 4-7, the results of the experiments conducted in this section are presented for the two classes VEB+ and SVEB+ introduced in section 1-8. These results are obtained by transforming the confusion matrix of the four AAMI classes into the confusion matrix of the two mentioned classes, and no changes have been made to

the models, and no further experiments have been performed. The best retrieval value for the VEB+ class was obtained by training the model on the balanced dataset using ADASYN. After that, the proposed under-sampling and oversampling methods rank second and third, respectively (Fig. 4-2). The proposed oversampling method has increased the retrieval value of the VEB+ class by 31.67% compared to the reference training set. The performance of the proposed under-sampling method is almost similar to the proposed oversampling method in all metrics. However, in this method, the training speed of the network is much higher than all data augmentation methods and even the reference case, because the number of samples that the model trains on in each iteration is significantly less (50,991 samples in the reference case, 183,380 samples in the augmentation case, and 8,146 samples in the proposed under-sampling case). This can contribute to speeding up the development of new methods. The proposed under-sampling method increased the retrieval value of the VEB+ class by 36.67%.

Table 4-7: Results of the experiments conducted in the Inter-Patient pattern for the VEB+ and SVEB+ classes.

Total				VEB+			SVEB+				
ACC	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**Description**	**Row**
0.59	0.74	0.78	0.71	0.50	0.44	0.57	0.97	0.96	0.98	Reference Dataset	**1**
0.91	0.69	0.77	0.63	0.39	0.28	0.62	0.95	0.93	0.98	Class Weight	**2**
0.73	0.57	0.61	0.53	0.14	0.08	0.44	0.84	0.74	0.97	SMOTE DA	**3**
0.80	0.65	0.76	0.56	0.24	0.14	0.76	0.88	0.80	0.98	SVMSMOTE DA	**4**
0.72	0.66	0.79	0.56	0.22	0.13	0.86	0.83	0.71	0.99	ADASYN DA	**5**
0.84	0.61	0.69	0.56	0.22	0.14	0.56	0.91	0.85	0.97	Traditional Under-sampling	**6**
0.88	**0.71**	**0.84**	**0.61**	**0.36**	**0.24**	**0.83**	**0.93**	**0.88**	**0.99**	**Proposed Oversampling DA**	**7**
0.83	**0.69**	**0.83**	**0.59**	**0.31**	**0.19**	**0.76**	**0.91**	**0.83**	**0.99**	**Proposed Under-sampling**	**8**

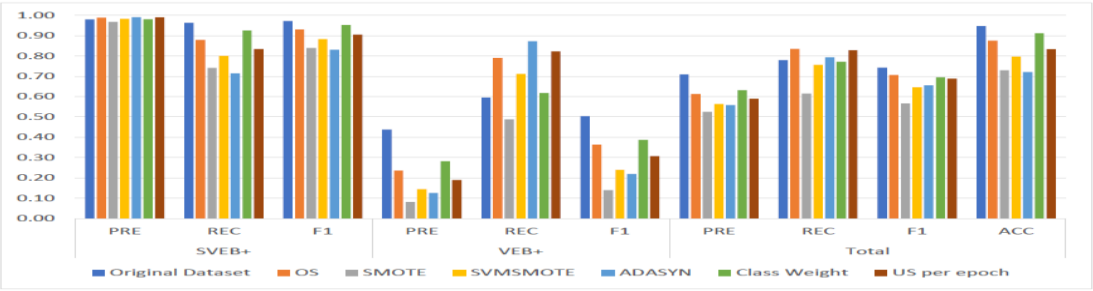

Fig. 4-2: Results of the experiments conducted in the Inter-Patient pattern for the two classes SVEB+ and VEB+.

4-4-2- Model Evaluation in the Intra-Patient Pattern.

The model is evaluated in the Intra-Patient pattern. In the Intra-Patient pattern, the proposed model is tested on a reference training set (imbalanced) and a balanced set using techniques such as SMOTE, SVMSMOTE, ADASYN, proposed oversampling, and traditional under-sampling. The performance of the model in terms of F1 score and accuracy is separately evaluated on the evaluation set, as shown in Tables 4-8. Additionally, precision and recall metrics for the four classes N, V, S, and F are listed in Table 4-10. The confusion matrix for the reference experiment, proposed oversampling, and proposed under-sampling is presented in Table 4-9. Based on these tables, the following conclusions can be drawn:

1. According to Table 4-8, data augmentation techniques did not have any significant impact on F1 score and accuracy. However, both traditional and proposed under-sampling methods decreased the overall F1 score by 19.35% and 6.45% respectively, and decreased precision by 5.05% and 2.02% compared to the reference case.

2. According to Table 4-10, the only method that improved the recall for the F class is the proposed oversampling method, which increased it from 0.77% to

0.91%. The precision and recall for classes N, V, and S did not change significantly with data augmentation methods, but the under-sampling methods had a negative impact on these metrics.

Table 4-8: F1 scores and the accuracy of the combined CNN model in the four AAMI classes in the Intra-Patient pattern.

Improvement(%)		Total		F	S	V	N		
ACC	**F1**	**ACC**	**F1**	**F1**	**F1**	**F1**	**F1**	**Description**	**Row**
-	-	0.99	0.93	0.84	0.97	0.91	1.00	Reference Dataset	**1**
-	-1.07	0.99	0.92	0.82	0.97	0.91	1.00	SMOTE DA	**2**
-	-	0.99	0.93	0.83	0.97	0.91	1.00	SVMSMOTE DA	**3**
-	-	0.99	0.93	0.84	0.97	0.91	1.00	ADASYN DA	**4**
-5.05	-19.35	0.94	0.75	0.38	0.84	0.63	0.97	Traditional Under-sampling	**5**
-	**-1.07**	**0.99**	**0.92**	**0.81**	**0.97**	**0.92**	**1.00**	**Proposed Oversampling DA**	**6**
-2.02	**-6.45**	**0.97**	**0.87**	**0.78**	**0.92**	**0.74**	**0.98**	**Proposed Under-sampling**	**7**

Table 4-9: Confusion matrix in (a) reference training set, (b) oversampled dataset, and (c) under-sampled dataset (pattern: Intra-Patient).

(A)

Actual \ Predicted	N	V	S	F
N	44890	72	73	6
V	130	1233	25	1
S	49	14	3421	20
F	55	0	36	310

(B)

Actual \ Predicted	N	V	S	F
N	44767	96	80	98
V	94	1280	13	2
S	51	26	3389	38
F	27	0	8	366

(C)

Actual \ Predicted	N	V	S	F
N	43754	781	480	26
V	250	1291	43	0
S	25	28	3447	4
F	67	7	50	277

Table 4-10: Proposed Combined Precision (PRE) and CNN Retrieval (REC) in the Intra-Patient Pattern for Four AAMI Classes.

Total		F		S		V		N			
REC	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**Description**	**Row**
0.91	0.95	0.77	0.92	0.98	0.96	0.89	0.93	1.00	0.99	Reference Dataset	**1**

0.94	0.91	0.85	0.79	0.97	0.97	0.94	0.88	0.99	1.00	SMOTE DA	**2**
0.93	0.92	0.84	0.82	0.96	0.98	0.94	0.87	0.99	1.00	SVMSMOTE DA	**3**
0.92	0.94	0.84	0.84	0.96	0.98	0.89	0.94	1.00	0.99	ADASYN DA	**4**
0.87	0.66	0.82	0.25	0.78	0.90	0.91	0.48	0.95	0.99	Traditional Under-sampling	**5**
0.95	**0.90**	**0.91**	**0.73**	**0.97**	**0.97**	**0.92**	**0.91**	**0.99**	**1.00**	**Proposed over-sampling DA**	**6**
0.89	**0.84**	**0.69**	**0.90**	**0.98**	**0.86**	**0.93**	**0.61**	**0.97**	**1.00**	**Proposed Under-sampling**	**7**

3- As mentioned in Section 2.2, in the Intra-Patient pattern, the model is familiarized with unique features of all records during training, which means the imbalance issue of the training set has a greater impact than the lack of information. Thus, augmenting the training set and preventing model bias towards class N while maintaining the F1 score of other classes improves their recall. In other words, since the number of class N data points is large and the Intra-Patient model can utilize information from all patients, it gains a relatively good understanding of normal heartbeats of different individuals. Therefore, even when the model is not biased towards this class (due to the augmented training set), it can correctly identify the majority of these normal heartbeats (Table 4-9). Consequently, FP remains low for other classes compared to the baseline experiment, resulting in little decrease in accuracy for classes V, S, and F.

4- In the Intra-Patient model, the sample reduction methods have resulted in a decrease in overall model performance in all four measures. In Table 4-9, the confusion matrix of the reference experiments shows the proposed increase and decrease in samples in the Intra-Patient pattern. In the reference experiment of this pattern, 130 and 55 samples from classes V and F, respectively, have been misclassified as class N. These values have decreased to 94 and 27 samples, respectively, in the sample increase experiment. However, the number of misclassified normal beats in these two classes has not changed significantly (considering the total number of normal samples).

The results in the VEB+ and SVEB+ classes are presented in Table 4-11 and Fig. 4-3. In this case, similar to the classification in the four AAMI classes, the data augmentation methods have no effect on F1 and overall accuracy. The proposed sample augmentation methods, SMOTE and SVMSMOTE, have increased the recall of the VEB+ class and the precision of the SVEB+ class while maintaining a constant F1 and overall accuracy. The proposed sample augmentation and reduction methods have increased the recall for the VEB+ class by 6.98% and the precision for the SVEB+ class by 2.33%, respectively.

Table 4-11: Results of experiments conducted in the Intra-Patient pattern in the VEB+ and SVEB+ classes.

Total				VEB+			SVEB+				
ACC	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**Description**	**Row**
0.99	0.95	0.93	0.96	0.90	0.86	0.93	1.00	1.00	0.99	Reference Dataset	**1**
0.99	0.94	0.96	0.93	0.89	0.92	0.86	1.00	0.99	1.00	SMOTE DA	**2**
0.99	0.94	0.96	0.93	0.89	0.92	0.86	1.00	0.99	1.00	SVMSMOTE DA	**3**
0.99	0.95	0.94	0.96	0.90	0.88	0.92	1.00	1.00	1.00	ADASYN DA	**4**
0.98	0.89	0.91	0.86	0.78	0.84	0.73	0.99	0.99	0.99	Traditional Under-sampling	**5**
0.99	**0.94**	**0.96**	**0.93**	**0.89**	**0.92**	**0.86**	**1.00**	**0.99**	**1.00**	**Proposed Oversampling DA**	**6**
0.98	**0.87**	**0.93**	**0.82**	**0.75**	**0.88**	**0.65**	**0.99**	**0.98**	**1.00**	**Proposed Under-sampling**	**7**

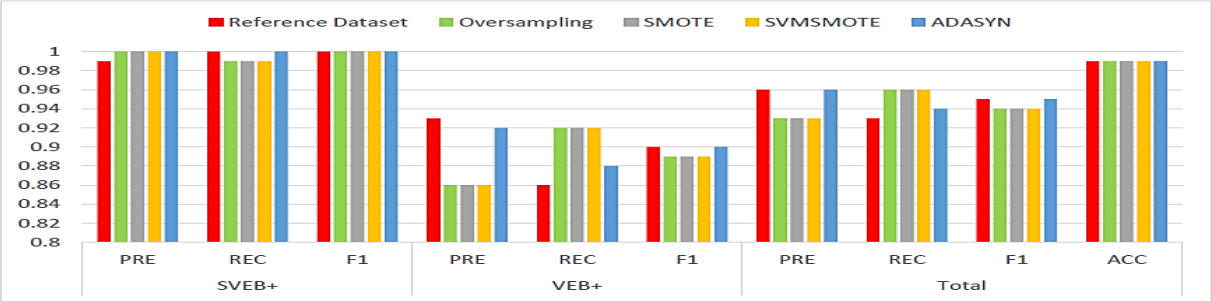

Fig. 4-3: Results of experiments conducted in the Intra-Patient pattern for two classes, VEB+ and SVEB+.

4-4-3- Evaluation of the Patient-Specific Model in the Pattern

In this pattern, a general model is trained on the general reference (non-equilibrium) training sets and balanced using the two proposed methods of oversampling and under-sampling. Then, in each of these three experiments, the general model is independently trained and evaluated on each of the 22 training-evaluation record sets described in Section 4-3. The results are presented in Table 4-12 for the F1 and accuracy metrics and in Table 4-13 for the precision and recall metrics for the N, V, S, and F classes in the three database conditions: reference database, under-sampled, and oversampled. Table 4-14 presents the results of the experiments for the two classes VEB+ and SVEB+ in the reference database, under-sampled, and oversampled conditions for the four metrics: accuracy, recall, F1, and precision. Based on the obtained results, in this pattern, whether the training set is balanced or unbalanced does not significantly affect the overall results. In the case where the general model is trained on the reference training set, the recall value for the SVEB+ class is 0.77, which increases to 0.82 in the oversampling experiment. Considering the importance of correctly detecting this type of beat, this finding can be desirable and significant.

Table 4-12: F1 and accuracy of the conducted experiments in the Patient-Specific pattern for the four classes N, V, S, and F.

Improvement(%)		Total		F	S	V	N		
ACC	**F1**	**ACC**	**F1**	**F1**	**F1**	**F1**	**F1**	**Description**	**Row**
-	-	0.99	0.91	0.80	0.97	0.86	0.99	Reference Dataset	**1**
-1.01	**0**	**0.98**	**0.91**	**0.82**	**0.97**	**0.84**	**0.99**	**Proposed Under-sampling DA**	**2**
-1.01	**-1. 09**	**0.98**	**0.90**	**0.81**	**0.96**	**0.84**	**0.99**	**Proposed Over-sampling DA**	**3**

Table 4-13: Precision and retrieval of experiments conducted in the Patient-Specific model across four classes N, V, S, and F.

Total		F		S		V		N			
REC	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**Description**	**Row**
0.88	0.94	0.80	0.80	0.97	0.98	0.76	0.99	1.00	0.99	Reference Dataset	**1**
0.88	**0.94**	**0.83**	**0.81**	**0.96**	**0.98**	**0.73**	**0.99**	**1.00**	**0.99**	**Proposed Under-sampling DA**	**2**
0.89	**0.91**	**0.75**	**0.04**	**0.86**	**0.97**	**0.84**	**0.84**	**0.99**	**0.99**	**Proposed Oversampling DA**	**3**

Table 4-14: Results of experiments conducted in the Patient-Specific pattern for the VEB+ and SVEB+ classes.

	Total				VEB+			SVEB+				
ACC	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**Description**	**Row**
0.99	0.92	0.88	0.97	0.85	0.77	0.95	0.99	1.00	0.99	Reference Dataset	**1**

0.99	**0.92**	**0.87**	**0.97**	**0.84**	**0.75**	**0.96**	**0.99**	**1.00**	**0.99**	**Proposed Under-sampling**	**2**
0.98	**0.91**	**091**	**0.92**	**0.83**	**0.82**	**0.85**	**0.99**	**0.99**	**0.99**	**Proposed Over-sampling**	**3**

4-4-4- Comparing three construction patterns for creating training and evaluation datasets.

To compare the three patterns of dataset creation, training, and evaluation, the results obtained from the proposed hybrid CNN model in experiments based on three training patterns, namely Inter-Patient, Intra-Patient, and Patient-Specific, are presented in Table 4-14. As expected, the results of the Intra-Patient pattern are significantly better than those of the Inter-Patient pattern. For example, in the reference experiment, the F1 and accuracy values for the Inter-Patient pattern are 0.69 and 0.94, respectively, while for the Intra-Patient pattern, they are 0.93 and 0.99, respectively. The Patient-Specific pattern also shows a noticeable superiority over the Inter-Patient pattern, with results close to the Intra-Patient pattern as well (Fig. 4-4). Similar trends can be observed in the classification of beats into the VEB+ and SVEB+ classes (Table 4-16). Considering these results and the fact that the Intra-Patient pattern is not easily implementable in practice, the utilization of the Patient-Specific pattern in developing automated arrhythmia detection systems tailored to each individual provides significant help in improving their performance.

Table 4-15: Comparison of results for the Inter-Patient, Intra-Patient, and Patient-Specific patterns in the classification of the four AAMI classes.

Total		**F**	**S**	**V**	**N**		
ACC	**F1**	**F1**	**F1**	**F1**	**F1**	**Description**	**Row**
0.94	0.69	0.52	0.64	0.50	0.98	Inter-Patient	**1**
0.99	0.93	0.84	0.97	0.91	1.00	Intra-Patient	**2**
0.99	0.91	0.80	0.97	0.86	0.99	Patient-Specific	**3**

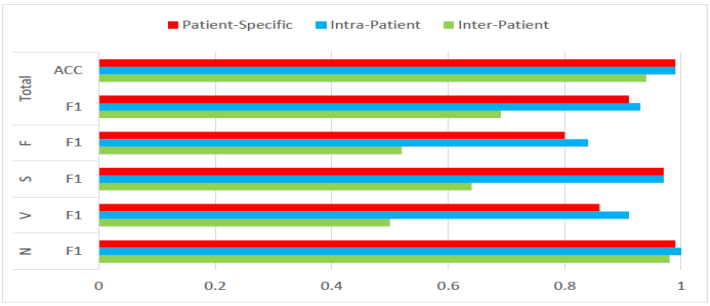

Fig. 4-4: Comparison of the results for three patterns.

Table 4-16: Comparison of results for three Inter-Patient, Intra-Patient, and Patient-Specific patterns in two classes VEB+ and SVEB+.

Total				VEB+			SVEB+				
ACC	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**Description**	**Row**
0.95	0.74	0.78	0.71	0.50	0.60	0.44	0.97	0.96	0.98	Inter-Patient	**1**
0.99	0.95	0.93	0.96	0.90	0.86	0.93	1.00	1.00	0.99	Intra-Patient	**2**
0.99	0.92	0.88	0.97	0.85	0.77	0.95	0.99	1.00	0.99	Patient-Specific	**3**

In order to evaluate the efficiency of the proposed architecture compared to the case where only deep learning is used, we re-evaluated the performance of the proposed model by removing the second sub-network (related to classical features). Additionally, we evaluated the performance of the proposed model in [7], which incorporates both classical features and deep learning in a conventional manner, on our own dataset. The results of these two experiments, along with the results obtained from the proposed model in the Intra-Patient pattern, are

shown in Table 4-17. Our proposed model achieves a higher F1 score compared to these two experiments, with an increase of 71.3% and 18.21% respectively.

Table 4-17: Comparison of the performance of the proposed model (row 3) to a model that solely utilizes deep learning (row 2) and a model that combines classical features and deep learning in a conventional manner (row 1).

Total				
ACC	**F1**	**REC**	**PRE**	**#**
98.35%	89.69%	89.68%	89.70%	1
96.55%	76.76%	68.66%	87.03%	2
99.04%	93.02%	90.84%	95.30%	3

4-5- The second model evaluation: Auto-Encoder Fusion

A proposed Auto-Encoder model has been applied to the Intra-Patient pattern on the reference training set and the balanced training set by increasing the samples, training, and separately evaluating them. The results of these experiments are presented in Table 4-18 and Table 4-19 for four classes (AAMI), and in Table 4-20 for the VEB+ and SVEB+ classes. Comparing the results in Table 4-18 and Table 4-19 demonstrates that the overall performance of this model in the reference training mode is lower compared to the previous proposed model (Combined CNN with Residual Structure). The use of increased samples has led to a decrease in F1 and accuracy in both classification scenarios for the four AAMI classes and the two VEB+ and SVEB+ classes. However, similar to the previous model, it has improved the recall for the V, S, F, and VEB+ classes. According to Table 4-21, the results of this model are inferior to the results of the first model (Combined CNN) but this model is much simpler and requires fewer computational and memory resources for implementation. Therefore, it can easily be used in wearable devices.

Table 4-18: F1 Score and Accuracy of the Integrated Auto-Coder-Encoder Model in Four AAMI Classes.

Total		F	S	V	N		
ACC	**F1**	**F1**	**F1**	**F1**	**F1**	**Description**	**Row**
0.98	0.84	0.61	0.93	0.84	0.99	Reference Dataset	**1**
0.96	**0.78**	**0.24**	**0.96**	**0.82**	**0.98**	**Proposed Oversampling DA**	**2**

Table 4-19: Precision and Recall of the Integrated Auto-Coder-Encoder Model in Four AAMI Classes.

Total		F		S		V		N			
REC	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**REC**	**PRE**	**Description**	**Row**
0.80	0.89	0.53	0.72	0.90	0.96	0.78	0.90	1.00	0.98	Reference Dataset	**1**
0.85	**0.72**	**0.61**	**0.15**	**0.97**	**0.95**	**0.87**	**0.77**	**0.96**	**0.99**	**Proposed Over-sampling DA**	**2**

Table 4-20: Results of the Integrated Auto-Coder-Encoder Model in VEB+ and SVEB+ Classification.

Total				VEB+			SVEB+				
ACC	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**F1**	**REC**	**PRE**	**Description**	**Row**
0.99	0.90	0.86	0.93	0.80	0.73	0.88	0.99	1.00	0.99	Reference Dataset	**1**
0.96	**0.80**	**0.89**	**0.73**	**0.59**	**0.82**	**0.46**	**0.98**	**1.96**	**0.99**	**Proposed Over-sampling DA**	**2**

4 - 6 - Comparison of proposed methods with previous research

Table 4-21 compares the best results obtained from two proposed models in the Intra-Patient pattern with previous works conducted in this pattern. Our proposed method performs better than the majority of current works by combining classical features with deep learning in a hybrid model. Additionally, most studies conducted in this pattern usually use over 80% of the available data for training their models and the rest for evaluation. However, here we have divided the data into 50% for training and 50% for evaluation. As a result, our model has been trained on a significantly smaller dataset and evaluated on a much larger one. To validate the proposed architecture of separating classical features from raw data in the hybrid approaches, the results of the proposed hybrid architecture are compared with the results of Model 9 (Row 8 in Table 4-21), which, as explained in section 2-3-7, utilized a bidirectional LSTM network for classification based on a combined feature vector composed of raw data and classical features. The proposed hybrid CNN model (Row 19 in Table 4-21) increased the overall retrieval rate by 73.14%. Moreover, the proposed hybrid encoder model (Row 21 in Table 4-21), although having a simpler structure compared to Model 9, increased the overall retrieval rate by 22.1%. In table 4-22, the results of our proposed hybrid CNN model have been compared with existing works in the Patient-Specific pattern. Our proposed method has achieved the best retrieval and F1 scores. In this pattern, row 1 of table 4-22 (row 54) has employed the conventional hybrid architecture. The proposed hybrid CNN model (row 13 of table 4-22) has improved the overall accuracy, retrieval, and F1 measures by 0.13%, 10.42%, and 6.10% respectively compared to model 54.

Table 4-21: Comparison of the proposed method with previous works in the Intra-Patient pattern.

ACC %	F1 %	REC %	PRE %	Model	The ratio of the number of samples of the training set to the total samples	Number of classes	Work	Row
99.39	-	-	-	Wavelet-LSTM	-	5	[43]	**1**
95.20	92.45	93.52	92.52	CNN	70%	13	[44]	**2**
85.00	-	90.00	-	CFT, FFT - DNN	-	5	[42]	**3**
98.00	93.69	97.70	90.00	CNN	21%	5	[37]	**4**
93.40	-	-	-	CNN	-	5	[51]	**5**
99.9	98.80	98.35	99.25	CNN	81%	2	[30]	**6**
94.20	-	95.30	-	DNN	80%	4	[50]	**7**
99.49	-	79.18	-	CFE - BiLSTM	90%	5	[9]	**8**
91.66	-	88.29	-	CNN	97%	4	[2]	**9**
99.04	**93.02**	**90.84**	**95.30**	**Hybrid CNN**	**50%**	**4**	**Proposed**	**19**
99.29	**94.66**	**93.01**	**96.37**	**Hybrid CNN**	**50%**	**2**	**Proposed**	**20**
97.92	**84.38**	**80.15**	**89.08**	**Hybrid AE**	**50%**	**4**	**Proposed**	**21**
98.68	**89.75**	**86.43**	**93.33**	**Hybrid AE**	**50%**	**1**	**Proposed**	**22**

Table 4-22: Comparison of results obtained by the proposed method with other studies conducted on the Patient model.

ACC %	F1 %	REC %	PRE %	Model	Number of classes	Work	Row
98.75	87.15	79.95	96.95	Wavelet-LSTM	2	[54]	**1**

97.50	-	63.34	-	CFE - DNN	5	[28]	**2**
97.90	88.00	80.20	97.30	SNN	4	[53]	**3**
97.50	85.14	85.90	84.40	CFE - DNN	4	[4]	**4**
98.75	**90.87**	**88.12**	**93.80**	**Hybrid CNN**	**4**	**Proposed**	**12**
98.75	**92.47**	**88.28**	**97.08**	**Hybrid CNN**	**2**	**Proposed**	**13**

5- Conclusion

In this paper, first, the importance of timely diagnosis of cardiac arrhythmias in detecting heart diseases and preventing sudden cardiac death caused by them was discussed, along with an introduction to cardiac arrhythmias and electrocardiogram (ECG) analysis. The existing challenges in automated arrhythmia diagnosis from ECG signals were described, and a comprehensive review of previous methods for ECG analysis was presented, categorized into three groups: classical methods, feature-based automatic methods, and a combination of classical and automatic features. Furthermore, a novel architecture for combining classical features with deep learning based on independent sub-networks, each suitable for a specific type of input, was introduced. Based on this architecture, two new models were proposed: a combined CNN model with a residual structure and a combined auto-encoder model. The impact of imbalanced training datasets on the model's performance was also explored using several methods for sample reduction and augmentation. Most of the examined balancing methods prevented model bias towards the majority class (normal) and improved the recovery of other classes, which can be clinically significant considering the importance of these classes. The performance of the proposed models was evaluated in three patterns: Inter-Patient, Intra-Patient, and Patient-Specific. Although the best results were obtained in the Intra-Patient pattern, as mentioned in Section 2, the results of this pattern are not practically significant. Based on the obtained results, the Patient-Specific pattern, by semi-automating the classifier, significantly improves its efficiency. This indicates that the use of the Patient-Specific pattern in automatic arrhythmia detection devices leads to overall performance enhancement. To validate the results of the proposed architecture properly, two additional experiments were conducted. One experiment utilized the proposed combined CNN model without the use of the sub-network related to classical feature analysis, as a model that solely relies on automated feature extraction. The other experiment used a model that combines classical features and deep learning in the conventional way (unified feature vector). The results showed that our proposed architecture-based combined CNN model outperformed both other models. In the end, the results obtained from the two proposed models were compared with the latest research findings in this field. The combined CNN model with a residual structure achieved better results than the latest research findings in this area. It is worth mentioning that this improvement in performance has been achieved without adding any significant processing overhead compared to previous methods. This is because classical features used in classification have been employed, which were already extracted in previous stages (such as heartbeat segmentation). Additionally, since the number of these classical features is small (only 5 classical features are used in the proposed models), the model has not become significantly more complex compared to a scenario where only deep learning is utilized. The proposed combined encoder model aims to provide a simple implementation of a combined architecture that can be used in wearable devices. The results obtained from this model can be compared with recent research findings. Therefore, this model demonstrates that the introduced combined architecture in this paper can achieve good results even in simple models with a small number of layers.

REFERENCES

[1] *SoftMax*. Available: https://towardsdatascience.com/softmax-activation-function-explained-a7e1bc3ad60

[2] Catalani, "Arrhythmia classification from ECG signals," Master, Engineering in Computer Science, Sapienza – University of Rome, Academic Year 2018 / 2019.

[3] Y. Hannun *et al.*, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," *nature medicine*, vol. 25, no. 1, pp. 65--69, 2019.

[4] K. Luo, J. Li, Z. Wang, and A. J. J. o. h .e. Cuschieri, "Patient-specific deep architectural model for ECG classification," *Journal of healthcare engineering*, vol. 2017, 2017.

[5] S. S. Virani *et al.*, "Heart disease and stroke statistics—2020 update: a report from the American Heart Association ",*Circulation*, pp. E139-E596, 2020.

[6] E. J. Benjamin *et al.*, "Heart disease and stroke statistics—2018 update: a report from the American

Heart Association," *Circulation*, vol. 137, no. 12, pp. e67--e492, 2018.

[7] G. Sannino and G. J. F. G. C. S. De Pietro, "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection," *Future Generation Computer Systems*, vol. 86, pp. 446-455, 2018.

[8] M. Romanò, *Text Atlas of Practical Electrocardiography: A Basic Guide to ECG Interpretation*. Springer, 2015.

[9] R. Li, X. Zhang, H. Dai, B. Zhou, and Z. J. I. A. Wang, "Interpretability Analysis of Heartbeat Classification Based on Heartbeat Activity's Global Sequence Features and BiLSTM-Attention Neural Network," *IEEE Access*, vol. 7, pp. 109870 - 109883, 2019 .

[10] Q. Yao, R. Wang, X. Fan, J. Liu, and Y. Li, "Multi-class Arrhythmia detection from 12-lead varied-length ECG using Attention-based Time-Incremental Convolutional Neural Network," *Information Fusion*, vol. 53, pp. 174-182, 2020.

[11] S. Hong, Y. Zhou, J. Shang, C. Xiao, and J. Sun, "Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review," *Computers in Biology and Medicine*, p. 103801, 2020.

[12] *Heartbeat and ECG*. Available: https://en.wikipedia.org/wiki/Electrocardiography

[13] Z. Ebrahimi, M. Loni, M. Daneshtalab, and A. Gharehbaghi, "A review on deep learning methods for ECG arrhythmia classification," *Expert Systems with Applications: X*, p. 100033, 2020.

[14] Francois, "Deep learning with Python," ed: Manning Publications Company, 2017.

[15] Géron, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media, 2019.

[16] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[17] G. B. Moody, R. G. J. I. E. i. M. Mark, and B. Magazine, "The impact of the MIT-BIH arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45-50, 2001.

[18] L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," circulation, vol. 101, no. 23, pp .e215-e220, 2000.

[19] M. Llamedo and J. P. Martínez, "Heartbeat classification using feature selection driven by database generalization criteria," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 3, pp. 616-625, 2010.

[20] J. L. Halperin and R .G. Hart, "Atrial fibrillation and stroke: new ideas, persisting dilemmas," *Stroke*, vol. 19, no. 8, pp. 937-941, 1988.

[21] W. B. Kannel, R. D. Abbott, D. D. Savage, and P. M. McNamara, "Epidemiologic features of chronic atrial fibrillation: the Framingham study," *New England Journal of Medicine*, vol. 306, no. 17, pp. 1018-1022, 1982.

[22] P. A. Wolf, R. D. Abbott, and W. B. Kannel, "Atrial fibrillation as an independent risk factor for stroke: the Framingham Study," *Stroke*, vol. 22, no. 8, pp. 983-988, 1991 .

[23] Pourbabaee, M. J. Roshtkhari, K. J. I. T. o. S. Khorasani, Man,, and C. Systems, "Deep convolutional neural networks and learning ECG features for screening paroxysmal atrial fibrillation patients," *nature medicine*, vol. 48, no. 12, pp. 2095-2104 , 2018 .

[24] G. Moody, A. Goldberger, S. McClennen, and S. Swiryn, "Predicting the onset of paroxysmal atrial fibrillation: The Computers in Cardiology Challenge 2001," in *Computers in Cardiology 2001. Vol. 28 (Cat. No. 01CH37287)*, 2001, pp. 113-116: IEEE.

[25] L. Goldberger et al., "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals," circulation, vol. 101, no. 23, pp. e215-e220, 2000.

[26] H. Li and P. Boulanger, "A Survey of Heart Anomaly Detection Using Ambulatory Electrocardiogram (ECG)," *Sensors*, vol. 20, no. 5, p. 1461, 2020.

[27] R. A. Harrigan, T. C. Chan, and W. J. Brady, "Electrocardiographic electrode misplacement, misconnection, and artifact," *The Journal of emergency medicine* ,vol. 43, no. 6, pp. 1038-1044, 2012.

[28] M. K. Das and S. J. I. s. r. n. Ari, "ECG beats classification using mixture of features," *International scholarly research notices*, vol. 2014, 2014.

[29] S. D. Goodfellow, A. Goodwin, R. Greer, P. C. Laussen, M .Mazwi, and D. Eytan, "Towards understanding ecg rhythm classification using convolutional neural networks and attention mappings," in *Machine Learning for Healthcare Conference*, 2018, pp. 83-101.

[30] X. Xu and H. Liu, "ECG heartbeat classification using convolutional neural networks," *IEEE Access*, vol. 8, pp. 8614-8619, 2020.

[31] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, "Application of deep convolutional neural network for automated detection of myocardial infarction using ECG

signals," *Information Sciences,* vol. 415, pp. 190-198, 2017.

[32] J. Gao, H. Zhang, P. Lu, and Z. Wang, "An effective LSTM recurrent network to detect arrhythmia on imbalanced ECG dataset," *Journal of healthcare engineering,* vol. 2019, 2019.

[33] B. N. Singh and A. K. Tiwari, "Optimal selection of wavelet basis function applied to ECG signal denoising," *Digital signal processing,* vol. 16, no. 3, pp. 275-287, 2006.

[34] P. Cao *et al.*, "A novel data augmentation method to enhance deep neural networks for detection of atrial fibrillation," *Biomedical Signal Processing and Control,* vol. 56, p. 101675, 2020.

[35] L. Donoho, "De-noising by soft-thresholding," *IEEE transactions on information theory,* vol. 41, no. 3, pp. 613-627, 1995.

[36] L. Donoho and J. M. Johnstone, "Ideal spatial adaptation by wavelet shrinkage," *biometrika,* vol. 81, no. 3, pp. 425-455, 1994.

[37] M. Shaker, M. Tantawi, H. A. Shedeed, and M. F. Tolba, "Generalization of Convolutional Neural Networks for ECG Classification Using Generative Adversarial Networks," IEEE Access, vol. 8, pp. 35592-35605, 2020.

[38] N. V. Thakor, J. G. Webster, and W. J. Tompkins, "Estimation of QRS complex power spectra for design of a QRS filter," *IEEE Transactions on biomedical engineering,* no .11 , pp. 702-706, 1984.

[39] M. Alfaras, M. C. Soriano, and S. J. F. i. P. Ortín, "A fast machine learning model for ECG-based heartbeat classification and arrhythmia detection," *Frontiers in Physics,* vol. 7, p. 103, 2019.

[40] C. Liu, P. Li, C. Di Maria ,L. Zhao, H. Zhang, and Z. Chen, "A multi-step method with signal quality assessment and fine-tuning procedure to locate maternal and fetal QRS complexes from abdominal ECG recordings," *Physiological measurement,* vol. 35, no. 8, p. 1665, 2014.

[41] S. Alvarado, C. Lakshminarayan, and J. C. Principe, "Time-based compression and classification of heartbeats," IEEE transactions on biomedical engineering, vol. 59, no. 6, pp. 1641-1648, 2012.

[42] Baghdadi, "Intelligent Diagnosis of Cardiovascular Disease using ECG Signals," Master of Science, Digital Electronics, Sharif University of Technology, International Campus, Kish Island, 2018.

[43] Ö. J. C. i. b. Yildirim and medicine, "A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification," *Computers in biology and medicine,* vol. 96, pp. 189-202, 2018.

[44] Ö. Yıldırım, P. Pławiak, R.-S. Tan, U. R. J. C. i. b. Acharya, and medicine, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," *Computers in biology and medicine,* vol. 102, pp. 411-420, 2018.

[45] S. Mousavi and F. Afghah, "Inter-and intra-patient ecg heartbeat classification for arrhythmia detection: a sequence to sequence deep learning approach," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 1308-1312: IEEE.

[46] J. Jiang, H. Zhang, D. Pi, and C. Dai, "A novel multi-module neural network system for imbalanced heartbeats classification," *Expert Systems with Applications: X,* vol. 1, p. 100003, 2019.

[47] P. De Chazal, M. O'Dwyer, and R. B. J. I. t. o. b. e. Reilly, "Automatic classification of heartbeats using ECG morphology and heartbeat interval features," *IEEE transactions on biomedical engineering* ,vol. 51, no. 7, pp. 1196-1206, 2004.

[48] E. J. d. 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 Systems with Applications,* vol. 40, no. 9, pp. 3561-3573, 2013.

[49] S. M. Mathews, C. Kambhamettu, K. E. J. C. i. b. Barner, and medicine, "A novel application of deep learning for single-lead ECG classification," *Computers in biology and medicine,* vol. 99, pp. 53-62, 2018.

[50] N. Golpayegani, "Cardiac Arrhythmia Classification Using Neural Networks and a Fuzzy Combination of Wavelet Transforms mid Autoregressive Modeling," presented at the The third international conference on Intelligent Decision, TEHRAN, IRAN, 2018 .

[51] M. Kachuee, S. Fazeli, and M. Sarrafzadeh, "Ecg heartbeat classification: A deep transferable representation," in *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, 2018, pp. 443-444: IEEE.

[52] M. M. Al Rahhal, Y. Bazi, H. AlHichri, N. Alajlan, F. Melgani, and R. R. Yager, "Deep learning approach for active classification of electrocardiogram signals," *Information Sciences,* vol. 345, pp. 340-354, 2016.

[53] Amirshahi, M. J. I. t. o. b. c. Hashemi, and systems, "Ecg classification algorithm based on STDP and R-STDP neural networks for real-time monitoring on ultra low-power personal wearable devices," IEEE transactions on biomedical circuits and systems, vol. 13, no. 6, pp. 1483-1493, 2019.

[54] S. Saadatnejad, M. Oveisi, M. J. I. j. o. b. Hashemi, and h. informatics" ,LSTM-Based ECG Classification for Continuous Monitoring on Personal Wearable Devices," *IEEE journal of biomedical and health informatics*, vol. 24, no. 2, pp. 515--523, 2019.

[55] F. J. a. p. a. Hadaeghi, "Reservoir Computing Models for Patient-Adaptable ECG Monitoring in Wearable Devices," *arXiv preprint*, 2019.

[56] F. N. Hatamian, N. Ravikumar, S. Vesal, F. P. Kemeth, M. Struck, and A. Maier, "The Effect of Data Augmentation on Classification of Atrial Fibrillation in Short Single-Lead ECG Signals Using Deep Neural Networks," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 1264-1268: IEEE.

[57] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321-357, 2002.

[58] He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, 2008, pp. 1322-1328: IEEE.

[59] Goodfellow *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, pp. 2672-2680, 2014.

[60] M. Delaney, E. Brophy, and T. E. Ward, "Synpaper of Realistic ECG using Generative Adversarial Networks," arXiv preprint arXiv:1909.09150, 2019.

[61] Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.