IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 9, 2021, accepted July 19, 2021, date of publication July 21, 2021, date of current version July 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3098986

An Ensemble of Deep Learning-Based Multi-Model for ECG Heartbeats Arrhythmia Classification

EHAB ESSA[1,2] AND XIANGHUA XIE[1], (Senior Member, IEEE)
[1]Department of Computer Science, Swansea University, Swansea SA1 8EN, U.K.
[2]Department of Computer Science, Faculty of Computers and Information, Mansoura University, Mansoura 35516, Egypt

Corresponding author: Xianghua Xie (x.xie@swansea.ac.uk)

This work was supported by the *Sêr* Cymru COFUND Fellowship.

ABSTRACT An automatic system for heart arrhythmia classification can perform a substantial role in managing and treating cardiovascular diseases. In this paper, a deep learning-based multi-model system is proposed for the classification of electrocardiogram (ECG) signals. Two different deep learning bagging models are introduced to classify heartbeats into different arrhythmias types. The first model (CNN-LSTM) is based on a combination of a convolutional neural network (CNN) and long short-term memory (LSTM) network to capture local features and temporal dynamics in the ECG data. The second model (RRHOS-LSTM) integrates some classical features, i.e. RR intervals and higher-order statistics (HOS), with LSTM model to effectively highlight abnormality heartbeats classes. We create a bagging model from the CNN-LSTM and RRHOS-LSTM networks by training each model on a different sub-sampling dataset to handle the high imbalance distribution of arrhythmias classes in the ECG data. Each model is also trained using a weighted loss function to provide high weight for not sufficiently represented classes. These models are then combined using a meta-classifier to form a strong coherent model. The meta-classifier is a feedforward fully connected neural network that takes the different predictions of bagging models as an input and combines them into a final prediction. The result of the meta-classifier is then verified by another CNN-LSTM model to decrease the false positive of the overall system. The experimental results are acquired by evaluating the proposed method on ECG data from the MIT-BIH arrhythmia database. The proposed method achieves an overall accuracy of 95.81% in the ''subject-oriented'' patient independent evaluation scheme. The averages of F1 score and positive predictive value are higher than all other methods by more than 3% and 8% respectively. The experimental results show the superiority of the proposed method for ECG heartbeats classification compared to many state-of-the-art methods.

INDEX TERMS Electrocardiogram (ECG), CNN, LSTM, bagging, ensemble, deep learning.

I. INTRODUCTION

Heart arrhythmia is any disturbance of the normal heart rate where the heart may beat too slowly, too early, too fast, or irregularly. Arrhythmias may cause symptoms including feeling dizzy, palpitations, fainting, and shortness of breath. Many arrhythmias are not dangerous; however, some arrhythmias types such as atrial fibrillation, premature ventricular contractions, and excessive supraventricular ectopic are associated with many cardiovascular diseases such as stroke, cardiac arrest or heart failure [1]–[3]. According to the World Health Organization (WHO) [4], cardiovascular diseases are the main cause of global mortality (about 31% of the global mortality in 2016).

Electrocardiography (ECG) records the electrical activity of the heart measured by a set of electrodes (usually 10) attached to the patient's skin. It's a common and non-invasive diagnosis technique to detect heart problems such as arrhythmias. There are different types of ECG configuration [5]. For example, 2-lead ECG is used to closely monitor the patient for a long period of time through Holter monitor device. ECG recording can remain from 24 to 48 hours and needs to be

The associate editor coordinating the review of this manuscript and approving it for publication was Marco Anisetti.

examined by a cardiologist to detect any heart problems. This can be a tedious process and very time-consuming. Therefore, finding an automated solution for analyzing and diagnosing ECG waves is crucial.

A normal heartbeat on ECG has three main waves as shown in Figure 1: the P-wave, which reflects the atria depolarization; the QRS complex, which reflects the depolarization of the ventricles and has the largest amplitude at R point; and the T-wave, which reflects the repolarization of the ventricles [6]. P-wave follows by a flat line called the PR segment indicates that the electrical impulse moves to the ventricles. ST-segment comes after the QRS complex shows that ventricles are completely depolarized.

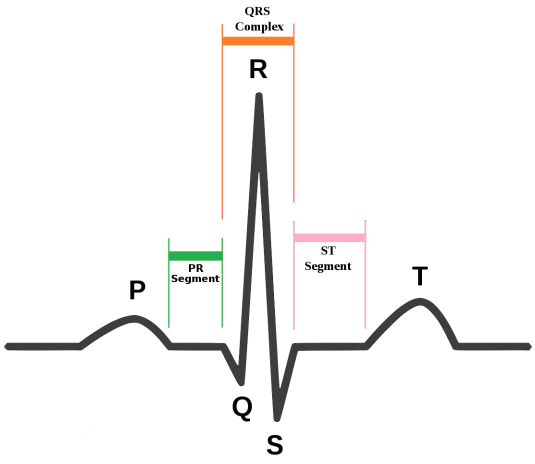


FIGURE 1. ECG wave of a normal heartbeat.

In this paper, we propose an ensemble of multi-model deep learning methods to automatically classify heartbeats arrhythmias in ECG data. Both convolutional neural networks (CNN) and long short-term memory (LSTM) are adapted in the proposed system. In the beginning, we define two different types of deep learning models. The first one is based on combining CNN and LSTM and the other one is extracting some classical features and fed them to the LSTM model. All these models are trained on bootstrap samples of the training data. LSTM has the ability to learn a temporal representation of the data. CNN can effectively extract local features from the raw sequential input. The additional classical features are less computationally expensive and help to boost the discrimination capability of the proposed system. Next, we introduced a deep learning network to fuse the result of all the bagging models trained in the first phase. Finally, in order to reduce the false positives, another CNN-LSTM model is presented. The proposed system is tested on the MIT-BIH arrhythmia dataset according to the recommendations of the Association for the Advancement of Medical Instrumentation (AAMI) [7]. The MIT-BIH dataset is heavily unbalanced making the recognition of arrhythmia classes a challenge. We address this problem by using the bagging technique to train the models on different distributions of the training data and reduce the imbalance level. Moreover, we modify the loss function by adding more weights to minority classes to encourage each model to correctly classify them. The main contributions of this paper can be summarized as:

- We propose multi-model deep learning method for automatic classification of ECG heartbeats arrhythmias by combining multiple deep learning models that are different in architecture and data-level to achieve a robust classification for the imbalance heartbeats data.
- A bagging of deep learning models has proposed based on two architectures: CNN-LSTM and LSTM combined with RR intervals and higher-order statistics (HOS) features to effectively classify all the heartbeat abnormality classes.
- A fusion classifier is proposed based on a meta-learning classification to dynamically formulate the final decision of the stack of deep learning models.
- A verification deep learning model is proposed to address the problem of false-positive classification of heartbeats arrhythmias in ECG signals.
- A weighted loss function is introduced for each deep learning model for better handling the imbalance in the data distribution.
- The proposed method is evaluated on the standard MIT-BIH database using a ''subject-oriented'' patient independent scheme that provides a realistic estimate of the classification performance.

The rest of the paper is organized as follows. In Section II, the related work of the ECG heartbeats arrhythmias classification is discussed. Section III presents the details of the proposed method. The experimental results and the ECG dataset are discussed in Section IV as well as a comparison with the state-of-the-art methods is presented. Finally, the conclusion is provided in Section V.

II. RELATED WORK

The classification of ECG arrhythmia has received much attention over the last two decades. According to the literature, there are three common steps: pre-processing, feature extraction, and classification. Feature extraction methods are mainly based on hand-crafted features such as morphological features [8]–[11], heartbeat intervals [8], [9], RR intervals [8]–[12], wavelet-based features [10]–[13], statistical features [10], [11], [14]–[16], and Hermite coefficients [14], [15]. However, hand-craft features like morphological features have a large inter-patient variation, so it is not enough to differentiate between different arrhythmia types. More recently, feature extraction based on deep learning techniques has emerged. For example, CNN has been used to extract features from raw waveform data as in [17]–[19]. In [20], CNN is combined with LSTM to analyze the ECG time series. Autoencoder is employed in [21], [22] to obtain deeply coded features. In [23], deep belief networks (DBN) is introduced to extract features from raw ECG data. Some other works try to hybrid both the hand-crafted features and deep learning techniques. For instance, combining wavelet-based and RR

intervals features with LSTM model [24], while in [25], authors integrate morphological features, RR intervals, and heartbeat intervals to DBN.

Many machine learning methods have been introduced to classify the features extracted from ECG signals such as support vector machine (SVM) [9], [11], [12], [14], [15], [21], [26], decision tree [10], linear discriminants (LDs) [8], AdaBoost [16], and deep learning methods [17]–[20], [22]–[25] e.g. CNN, LSTM, auto-encoder, DBN. However, there are some limitations on the existing methods such as many of these methods are not scalable enough to classify ECG records of new subjects due to its large variations that may have or carefully selected training/testing data or classes without following the well-known AAMI recommendations. Note that the works that do not follow the AAMI recommendations cannot be included in any comparison.

The MIT-BIH arrhythmia database is the standard test material for the performance evaluation of arrhythmia classification methods. There are two types of evaluation: ''class-oriented'' and ''subject-oriented'' [12]. In the ''class-oriented'' evaluation e.g. [14], [16], [18], [20], [22], [27], all the ECG records from all patients are put together and then split into training and testing sets. However, this kind of evaluation produces optimistic results that maybe not sensible in real applications, as the inter-patient variation is not considered due to the training and testing sets contain samples from the same patients.

In the ''subject-oriented'' evaluation, the dividing of training and testing sets is based on the ECG patient's records, where a single patient's records can only be in either training or testing sets. This type of evaluation maintains the inter-patient variation and provides a more practical evaluation of the performance of heartbeat classification methods. The ''subject-oriented'' evaluation can also be categorized into the patient-specific [13], [17], [23], [24] and patient-independent evaluation [8]–[12], [19], [25], [26], [28]. In the patient-specific scheme, a small part at the beginning of a particular patient's record with the annotated data is used to adapt a pre-trained classifier. This improves the performance of patient-specific classifiers than patient-independent. However, the cost of a patient-specific scheme is much higher since it may require intervention from an expert to label some heartbeats that making the entire approach is not practical and time-consuming. In this paper, we propose a deep learning patient-independent approach that follows the ''subject-oriented'' scheme.

Many patient-independent approaches have been introduced to classify heart arrhythmia, e.g. In [8], linear discriminants have been proposed to categorize ECG heartbeats. This method integrates a various set of features based on RR intervals, heartbeat intervals, and ECG morphology. In [12], RR intervals and morphological features extracted using wavelet transform and independent component analysis have been utilized to recognize irregular heartbeat. These features are used to train a set of SVM classifiers independently and then fused together to obtain the final decision. In [26], the least-square twin SVM classifier is proposed to classify sparse features computed over a Gabor dictionary. The learning parameters are optimized using particle swarm optimization (PSO). In [28], Herry *et al.* characterize the ECG signal by using heartbeats interval features and synchro-squeezing transform to analyze time-varying oscillatory patterns of heart rhythms and then classify these features by an SVM classifier into four classes. However, the accuracy of these methods is still restricted.

In an attempt to increase the ECG classification accuracy, different ensemble-based techniques have been used to create multiple instances and then combine them to produce a robust model. In [9], Zhang *et al.* train an ensemble of SVMs using a one-vs-one scheme to classify heartbeats. ECG morphology and intra- and inter-beat intervals are used as input features. The ensemble is made by using the product rule of all the created models. The method is extended in [11] by training each individual SVM model on one type of features and expanding the extracted features to include wavelets, HOS, and local binary patterns (LBP). In [10], Shi *et al.* use an extreme gradient boosting model to perform hierarchical classification of different heartbeats types. The method selects a subset of features from a pool of hand-crafted features. However, all these methods are mainly using hand-crafted features and not generalize enough to classify ECG arrhythmia.

In the last few years, deep learning has led to impressive performance in many applications such as object recognition due to its huge ability to detect salient patterns in the input data. In [19], authors train CNN on raw ECG data containing two consecutive heartbeats and add batch-weight loss function to handle the imbalance problem in the ECG dataset. However, the false-positive rate is significantly high and the accuracy of recognizing the normal heartbeats (i.e. the majority class) is considerably reduced. In [29], Jiang *et al.* proposed over-sampling the minority classes to overcome imbalance data and using auto-encoder as a feature extractor to be fed to CNN. However, the method becomes more computationally expensive, and this over-sampling strategy may also cause over-fitting. In [25], Mathews *et al.* use a stack of Restricted Boltzmann Machines (RBM) to encode the extracted hand-crafted features and form a deep belief network (DBN). However, the method is heavily dependent on the hand-crafted features which affect its generalization ability to classify ECG arrhythmia. In [23], authors extend the previous method by training the DBN on the raw ECG data. However, the method is not taking a benefit of the time-dependencies of ECG signal.

III. PROPOSED METHOD

Figure 2 shows an overall view of the proposed system. We propose a cascade of a multi-model deep learning-based ensemble to classify ECG heart arrhythmia by training the models of each heartbeats class sequentially using a one-vs-all scheme. The first step is pre-processing and segmenting the input ECG signals. Next, we create an ensemble of two deep learning models using the bagging technique.

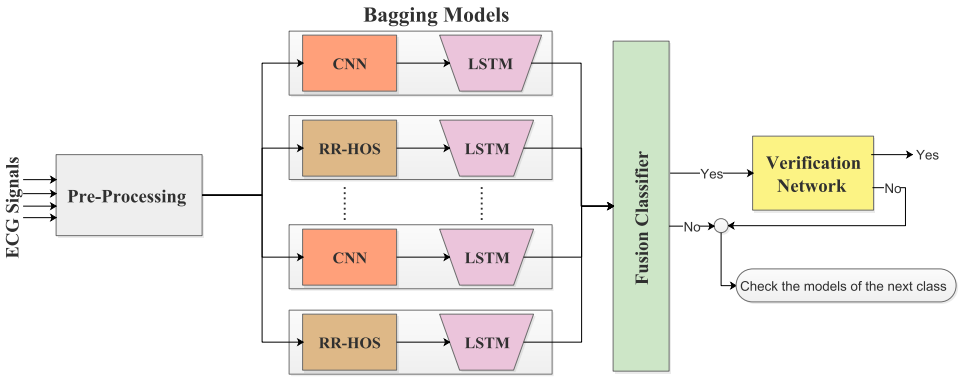


FIGURE 2. An overview of the proposed method.

The first model (CNN-LSTM) is defined based on CNN and LSTM, and the second model (RR-HOS-LSTM) is combining RR-intervals and HOS features with LSTM. Then, a meta-learning classifier is proposed to fuse the output of all bagging models. Finally, a verification network is introduced to further check the validity of the predicated class for the input ECG segment. The cascaded model is obtained by repeating the same procedure consecutively for each heartbeat class. If the input ECG heartbeat segment is classified as negative at any stage, it will be tested by the next set of models for the next heartbeat class.

A. PRE-PROCESSING AND HEARTBEATS SEGMENTATION

We apply two pre-processing steps as in literature [8], [9], [11] to remove the baseline wandering and reducing the high-frequency noise. First, the ECG baseline is obtained by utilizing two median filters of 200-ms and 600-ms, one after the other. The first median filter removes QRS complexes and P-waves, while the second one removes T-waves. To produce the baseline-corrected signal, the baseline is then subtracted from the raw ECG signal. Next, the power-line and high-frequency noise is removed by using a low-pass filter with a cut-off frequency of 35 Hz. The resulted signal will be employed as an input to the deep learning models. Figure 3 shows an example of ECG signal before and after the pre-processing steps.

The ECG data consists of a sequence of heartbeats. Many heartbeat segmentation algorithms have been proposed e.g. Pan and Tompkins used adaptive thresholding to detect QRS complex [30]. However, the main focus of this paper is on the classification, thus to obtain the heartbeats segment, the annotation of the QRS complex included with the MIT-BIH database was used. Here, a window of 180 samples is taken around the R-peak to represent the heartbeat.

B. CLASSICAL FEATURE EXTRACTION

The most common classical features used to describe ECG data is RR intervals [8]–[12], [24], [25]. RR intervals are defined as the time between R-peak points of successive heartbeats. Four RR intervals are usually computed:

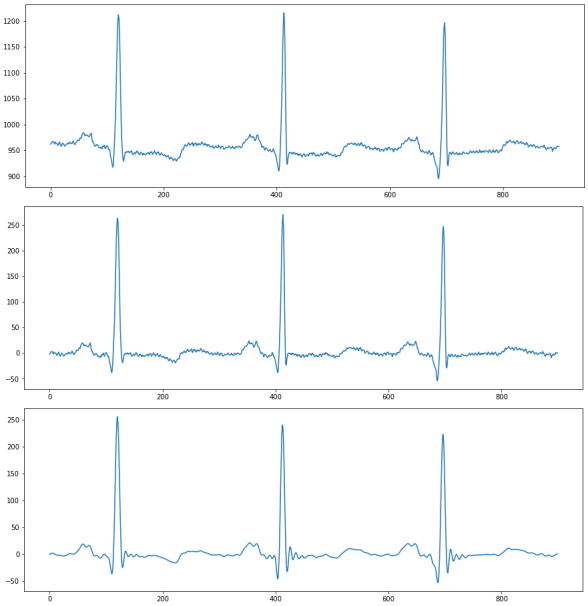


FIGURE 3. An example of ECG signal before and after pre-processing. The first row represents the raw ECG signal. The second row shows the ECG signal after removing the baseline wandering. The third row represents the ECG after removing the high-frequency noise.

pre-RR, post-RR, local-RR, and average-RR. Pre-RR is the RR interval between the desired heartbeat and the past one. Post-RR refers to the RR interval between the desired heartbeat and the following one. Local-RR indicates the mean of the past 10 RR intervals of a given heartbeat. Similarly, average-RR is the average of the past 5 min RR intervals of a given heartbeat. Moreover, the normalized version of the previous four intervals is also computed by dividing each feature on its global average with the same ECG record.

HOS has shown to be better than the morphological ECG [11], [15]. HOS indicates to skewness and kurtosis, which measures sharpness and asymmetry of the heartbeat. Here, the heartbeat is split to 6 intervals, and then the skewness and kurtosis are measured over each one.

C. DEEP FEATURE EXTRACTION

CNN and LSTM are exploited to highlight local and temporal features of the ECG signal. CNN has employed to extract spatial features, while LSTM has used to model long-term contextual dependencies. In this paper, both models are utilized to extract spatio-temporal features.

1) CNN

CNN is one of the successfully deep learning models in many fields, such as computer vision [31], [32], natural language processing [33], [34], and speech recognition [35]. Due to its high capability of extracting hierarchical feature representation, that is relatively robust to noise. It has some unique properties, such as weight sharing, local connectivity, and spatial pooling. The CNN has fewer parameters compared to the traditional feedforward network.

CNN not only can handle 2D/3D inputs such as image and video but also 1D sequences such as ECG signals. CNN mainly consists of a sequence of convolutional layers and pooling layers. Convolutional layers apply a set of learned filters over the whole input sequence to generate feature maps that highlight the existence of those features in the raw input sequence. The feature maps generated by the convolutional layers go through a non-linear transformation called an activation function. The output feature map x_j at layer ℓ is defined by convolving kernel k_i^ℓ with the input feature map $x_i^{\ell-1}$ as follows:

$$x_j^\ell = \sigma\left(\sum_i x_i^{\ell-1} * k_i^\ell + b_j^\ell\right) \tag{1}$$

where b_j^ℓ is the bias term, and σ is the activation function. There are many activation functions in the literature, such as sigmoid, tanh and rectified linear unit (ReLU). In the deep neural networks, the ReLU is the most common activation function because of its simplicity, and it is not saturated like sigmoid function so it can alleviate the problems of vanishing gradients. The pooling layer is applied to down-sample each feature map to produce less sensitive features to local translation and also reduce the number of model parameters. Max-pooling is the popular function used in the pooling layer, where the maximum value is computed for each window of the feature map. Max-pooling has the ability to extract the most prominent features. A stack of convolutional and pooling layers allows learning high-level features as a function of low-level features.

There are two additional layers that can be added to any deep neural networks; dropout and batch normalization. Dropout layer randomly removes a specified proportion of the connections between two consecutive layers during training the network in order to reduce the neural network over-fitting problem. Batch normalization layer is normalizing the input to a layer with the mean and standard deviation of the mini-batch.

2) LSTM

One of the most chosen architectures for modeling time series data is known to be the recurrent neural network (RNN). RNNs have been employed in many applications such as handwriting recognition [36], machine translation [37]. RNN varies in the learning structure from feedforward neural networks by connecting the outputs of the current time step to the inputs of the next time step. This allows the RNN to retain the internal state for the sequential input to be processed.

The traditional RNN has some limitations due to the problems of vanishing or exploding gradients that encounter when modeling long-term dependencies. Therefore, LSTM is proposed to overcome these limitations. The LSTM model replaces the RNN neurons in the hidden layer with LSTM neurons. LSTM has a special structure including a memory cell and three gate regulators. The memory cell maintains relative information about the dependencies between input elements all over the processing of the sequence. LSTM comprises three gates (input i_t, forget f_t, and output o_t gates) to regulate the stream of information. The input gate controls how much information to be stored in the memory cell. The output gate manages the output information from the memory cell C_t to the next LSTM unit. The output gate decides what the hidden state h_t should be. The forget gate allows the memory cell to be reset. The following equations define the process of the three gates and the memory cell of LSTM:

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$
$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$
$$o_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \tag{4}$$
$$\tilde{C}_t = \tanh(w_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{6}$$
$$h_t = \tanh C_t \cdot o_t \tag{7}$$

where x_t is the input at time t; w_i, w_f, w_o, w_C are the weights, b_i, b_f, b_o, b_C are the biases; σ, tanh are the sigmoid and hyperbolic tangent activation functions.

D. BAGGING-BASED DEEP LEARNING MODELS

In this work, an ensemble of different deep learning models is proposed to classify ECG heartbeats. The first model based on an aggregation of CNN and LSTM. The RRHOS-LSTM model combines the RR intervals and HOS features with LSTM. The ensemble is created by using a bagging technique to enhance the robustness of the model and address the data imbalance issue.

The ECG arrhythmia dataset (MIT-BIH) is highly imbalanced, where 89% of the training records are labeled as normal. This biased distribution of data highly affects the learning process of minority classes. To increase generalization capability, we used the bootstrap method to create small multiple subsets of data by randomly sampling with replacement. In the beginning, we convert the classification problem from a multi-class to binary using a one-vs-rest strategy. where the data from one class is labeled as positive

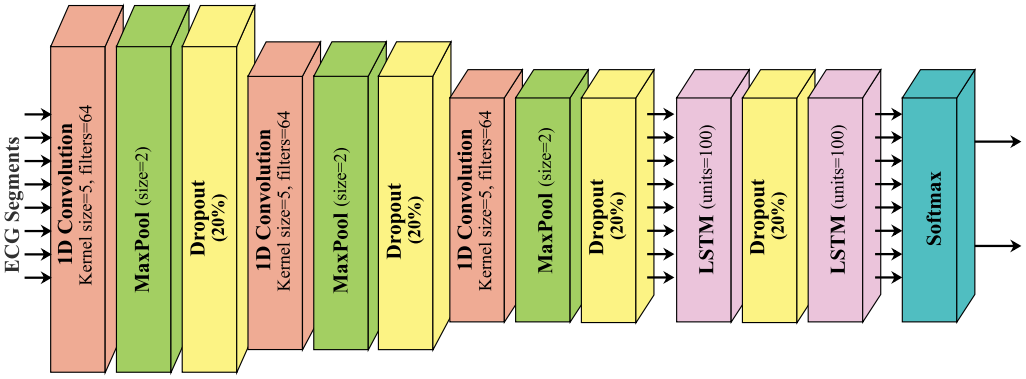


FIGURE 4. The architecture of the proposed CNN-LSTM with three convolutional, and max pooling layers and two LSTM layers.

and all the other data from the rest of the classes are negative. Then, the negative samples are randomly down-sampled for each binary class model while holding the positive samples the same to build the training set. This procedure is replicated several times to construct multiple training sets. The deep learning models are trained on these subsets of data in order to achieve a collection of different models in both data level and architecture.

The purpose of sub-sampling is not to get a balanced dataset since minority classes contain a few hundred samples compared to thousands of samples of the majority class. The sub-sampling is producing an imbalanced dataset with a pre-defined ratio between the negative and positive samples. This is to keep many useful negative samples to improve the overall accuracy of the model. The weighted loss function is also utilized here to hinder the model from learning only the majority class as discussed in Section III-G.

1) CNN-LSTM MODEL

The CNN-LSTM network, as shown in figure 4, includes convolutional layer and max-pooling repeated for 3 times, and followed by 2 LSTM layers. The input of this model comprises 5 consecutive heartbeats, where 2 previous and 2 successor heartbeats accompany the present heartbeat. The network input has a size of 900 (5×180). The input of filters set to 64 for all convolutional layers. The kernel size of each convolutional layer set to 5. The kernel moves one step (i.e. stride equal to 1) over the input sequence at a time and is convoluted with the corresponding input elements. After each convolutional layer, the ReLU activation function is applied. For each max-pooling, the pooling size set to 2 with stride 2. This cuts the output size of each layer by half. We introduce a dropout of 20% after each max-pooling layer to minimize the over-fitting of the network.

Two LSTM layers are the last configuration of the CNN-LSTM network, accompanied by a softmax layer to predict the output class. the LSTM layers derived the temporal dynamics from CNN's features. The number of hidden neurons of each LSTM unit is 100. The output of the hidden state of each neuron of the first LSTM layer is used as an input to the next LSTM layer. The last LSTM layer returns the last hidden state output which captures an abstract representation of the input sequence. The probability of the heartbeat output class is produced by the softmax function.

2) RRHOS-LSTM MODEL

The RRHOS-LSTM model blends HOS and RR intervals features with LSTM. As shown in Figure 5, the RRHOS-LSTM consists of a feature extraction layer followed by an LSTM. RR intervals are obtained to reveal global and local information regarding the R-peak of two sequential heartbeats The number of RR intervals features is 8 as described in Section III-B. HOS measures high-order statistical features from the input heartbeats. HOS extracts 60 features from five successive heartbeats, where 2 predecessor and 2 successor heartbeats accompany the current one. The number of the input features to the LSTM is 68. The input features are processed by the LSTM to learn temporal dynamics. A dropout layer of 20% is added after the LSTM to reduce the model over-fitting. The output hidden states for each time step of the LSTM are produced and given to a softmax layer to label the input ECG heartbeat.

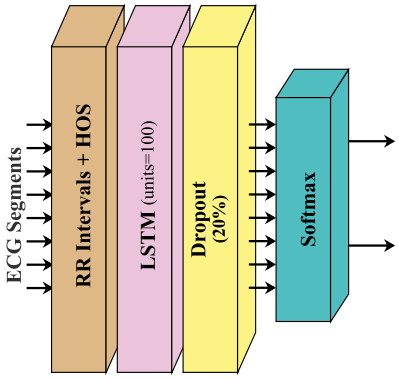


FIGURE 5. The proposed RRHOS-LSTM architecture with one feature extraction layer followed by one LSTM layer.

E. FUSION CLASSIFIER

The fusion classifier combines the output of all bagging models in order to form an ensemble model. The fusion classifier is a meta-learner that can correct the predictions from the base models and boost the overall system performance. It is a feedforward neural network that takes the probabilistic output of the RRHOS-LSTM and CNN-LSTM models as an input. The input size of the fusion classifier is $2 * N$, where N is the total number of the bagging models. The fusion classifier includes (see Figure 6) two fully-connected layers preceded by a batch normalization layer. The number of hidden neurons is 500 in each dense layer, and each neuron has ReLU activation function. Between the two dense layers, a dropout layer with a 20% ratio is added. A softmax function is used in the final layer to generate a probability distribution over predicted output classes.

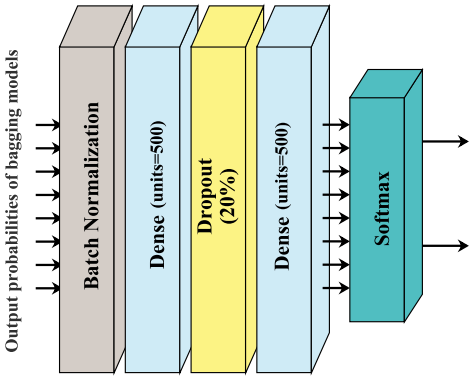


FIGURE 6. The proposed fusion classifier architecture with a batch normalization and two fully connected layers.

F. VERIFICATION NETWORK

Finally, a verification network is introduced to validate the performance of the fusion classier and decrease the false positive. It is a deep neural network based on the CNN and LSTM. The verification network (as shown in Figure 7) comprises 3 consecutive layers of convolution and max-pooling followed by two LSTM layers. Dropout regularization is the difference between the CNN-LSTM model described in Section III-D1 and the verification network. Two dropout layers are used in the verification network, the first one after the convolutional part and the second one after the LSTM layers. The dropout rate is set at 50% in order to reduce the network's over-fitting. Note, the RRHOS-LSTM is not used at this stage to reduce the model complexity and the over-fitting problem.

We repeat the same procedure consecutively for each heartbeat class to form a cascade model to accomplish the classification system. The proposed system has two binary models per class: the fusion classifier, which merges the output of bagging of CNN-LSTM and RRHOS-LSTM models, and the verification network. In the cascade scheme, suppose the number of heartbeat arrhythmia classes is C, we need to have $C - 1$ of the fusion classifiers and the same number for the verification networks. We exclude the normal class from having its own models so that if the testing sample is classified as negative by all binary models from the other classes then it is classified as a normal heartbeat. The testing samples go to the first fusion classifier representing the first class, and then transfer to the verification network if it is labeled as positive for confirmation. If it is classified as negative by one of the two models, then it moves to the subsequent fusion classifier and verification network and so on.

G. WEIGHTED LOSS FUNCTION

Since the ECG data distribution is highly imbalanced and performing over-sampling of the minority classes may cause over-fitting and increasing the time complexity of training the model, we proposed to use weighted loss function to force each model to give more attention to the minority classes. In this work, we use a weighted cross-entropy as a loss function with the softmax layer. The weighted cross entropy loss function $\mathcal{L}$ can be defined as:

$$\mathcal{L} = -\sum_{c=1}^{M} \beta_c y_{o,c} \log(p_{o,c}) \tag{8}$$

where β_c is the weight function for class c, $y_{o,c}$ is the groundtruth binary indicator, and $p_{o,c}$ is the predicted value for the observation o to be classified as c. The weight β_c is defined based on the frequencies of the samples as follows:

$$\beta_c = \frac{F_j}{F_c} \tag{9}$$

where F_j is the number of samples of the majority class and F_c is the number of sample for class c. The bootstrap procedure is down-sampling the majority class according to the specified ratio. Here, the ratio 1:4 is used, which means the number of negative samples is four times larger than the positive samples. The weighted cross-entropy loss function is important to make sure the model is learning the minority classes and not over-fitted by the majority class.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. ECG ARRHYTHMIA DATABASE

The MIT-BIH database [38], [39] is a typical ECG arrhythmia dataset for testing the efficiency of heartbeat classification approaches. The dataset contains 48 recordings obtained from 47 subjects (22 women and 25 men) over approximately 30 minutes. The data is sampled at 360 Hz and associated with the annotations of the R-peak location of each heartbeat. The database encloses about 109,000 beats. The annotations were carried out by multiple cardiologists independently, and all disputes were reviewed and resolved. Each ECG record composed of two-lead signals: the modified-lead II (MLII) and the other signal is one of lead V1, V2, V3, V4, or V5. In this work, MLII signals have only been used. According to AAMI recommended, four records with paced beats are excluded, named as 102, 104, 107, and 217. Originally, the MIT-BIH database classified the heartbeats to 16 types. These types of heartbeats are grouped into five categories

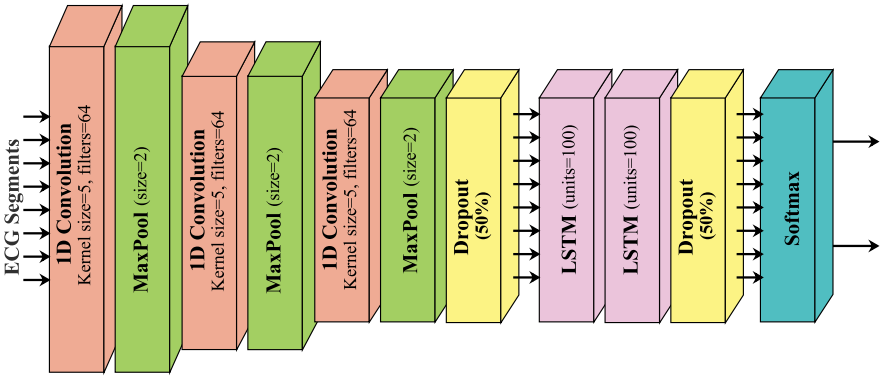


FIGURE 7. The proposed verification network architecture.

TABLE 1. Evaluation of different ratios using a bagging of CNN-LSTM classifiers.

	SVEB					VEB					F				
Bagging ratio	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc
1:2	86.08	76.74	13.74	23.70	77.13	93.69	98.53	81.56	87.21	98.22	25.77	96.28	5.17	8.62	95.72
1:4	81.63	84.43	18.41	30.04	84.31	94.50	98.74	83.88	88.87	98.47	45.10	93.35	5.08	9.13	92.97
1:6	83.54	81.46	16.25	27.20	81.55	95.02	97.53	72.69	82.37	97.36	58.76	81.10	2.39	4.60	80.92
1:8	43.38	85.40	11.34	17.98	83.67	95.21	98.75	84.03	89.27	98.52	35.31	90.66	2.89	5.35	90.22

according to AAMI recommends: Normal (N), Ventricular ectopic beat (VEB), Supraventricular ectopic beat (SVEB), Fusion (F), and Unknown beat (Q). The Q class is excluded here since it has a comparatively very limited number of samples (i.e. 12 samples). Here, we follow the evaluation scheme of ''subject-oriented'' patient independence. The database is partitioned into the training set (DS1) and test set (DS2) as in [8] to preserve inter-patient heterogeneity. Each dataset comprises 22 ECG recordings from different subjects with about the same number for each heartbeats type. This type of evaluation scheme allows comparing different types of heartbeats classification approaches in a fair. The training set (DS1) is split into two sets, one for training the bagging models and verification network which contains 90% of DS1, and the other 10% for training the fusion classifier.

B. IMPLEMENTATION DETAILS

We run all our experiments on a single GPU Nvidia TITAN Xp. TensorFlow library [40] is used to implement the proposed models. We used Adam optimizer for training the bagging models and RMSProp optimizer for the fusion and verification network. The learning rate of all classifiers is 0.001. The batch size of the bagging models is 256, while the fusion and verification network is 128. The batch normalization is used in the fusion classifier for all classes except for class F. All these parameters are optimized through a grid search on the training set.

C. EVALUATION METRICS

The evaluation of the proposed method's performance is carried out using 6 metrics. These metrics includes accuracy (Acc %), F1 score (F1 %), specificity (Sp %), sensitivity (Se %), positive predictive value (PPv %), and Cohen's Kappa (κ) and defined as follows:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}, \tag{10}$$

$$F1 = 2 \times \frac{PPv \times Sens}{PPv + Sens}, \tag{11}$$

$$Spec = \frac{TN}{TN + FP}, \tag{12}$$

$$Sens = \frac{TP}{TP + FN}, \tag{13}$$

$$PPv = \frac{TP}{TP + FP}, \tag{14}$$

$$\kappa = \frac{P_0 - P_e}{1 - P_e} \tag{15}$$

where TP refers to the number of heartbeats from the given class that are correctly classified. FP indicates the number of heartbeats misclassified as from the given class. FN is defined as the number of heartbeats originally from the given class but misclassified as from one of the other classes. Finally, TN indicates the total number of heartbeats that are correctly classified as from the other classes. All these metrics are derived from the confusion matrix. Cohen's Kappa measures agreement between two raters. P_0 is the observed agreement between the raters (i.e. overall accuracy), and P_e is the agreement by chance alone.

D. CLASSIFICATION PERFORMANCE

Firstly, we conduct an experiment for evaluating different sampling ratios of CNN-LSTM bagging classifiers. Table 1 reports the result of 4 sampling ratios for training

TABLE 2. Evaluation of different size of input heartbeats on CNN-LSTM bagging models using one, three, or five beats as an input and also RRHOS-LSTM bagging models using three or five beats.

	SVEB					VEB					F				
	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc
1 Beat CNN-LSTM	67.22	86.26	17.39	27.63	85.47	90.11	88.91	36.04	51.48	88.99	94.07	65.42	2.10	4.11	65.65
3 Beats CNN-LSTM	84.27	82.00	16.77	27.97	82.09	95.52	98.40	80.50	87.37	98.21	52.06	87.40	3.15	5.95	87.12
5 Beats CNN-LSTM	81.63	84.43	18.41	30.04	84.31	94.50	98.74	83.88	88.87	98.47	45.10	93.35	5.08	9.13	92.97
3 beats HOS+RR	82.62	66.33	9.56	17.13	67.00	94.31	96.20	63.21	75.69	96.07	92.53	78.71	3.31	6.40	78.81
5 beats HOS+RR	84.27	68.58	10.35	18.43	69.22	94.62	96.32	64.08	76.41	96.21	86.60	84.51	4.22	4.22	84.52

TABLE 3. A comparison of the performance of combining bagging models using the proposed fusion classifier and the majority voting and then validating the fusion results using the verification network.

	SVEB					VEB					F				
	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc
Fusion Classifier	80.12	86.67	20.55	32.71	86.40	95.62	99.38	91.41	93.47	99.13	38.40	98.18	14.26	20.80	97.71
Majority Voting	83.44	78.44	14.28	24.39	78.65	94.93	99.53	93.31	94.11	99.23	83.51	91.46	7.15	13.18	91.39
Verification Network	65.51	98.56	66.19	65.85	97.20	93.91	99.62	94.55	94.23	99.25	19.33	99.79	41.67	26.41	99.16

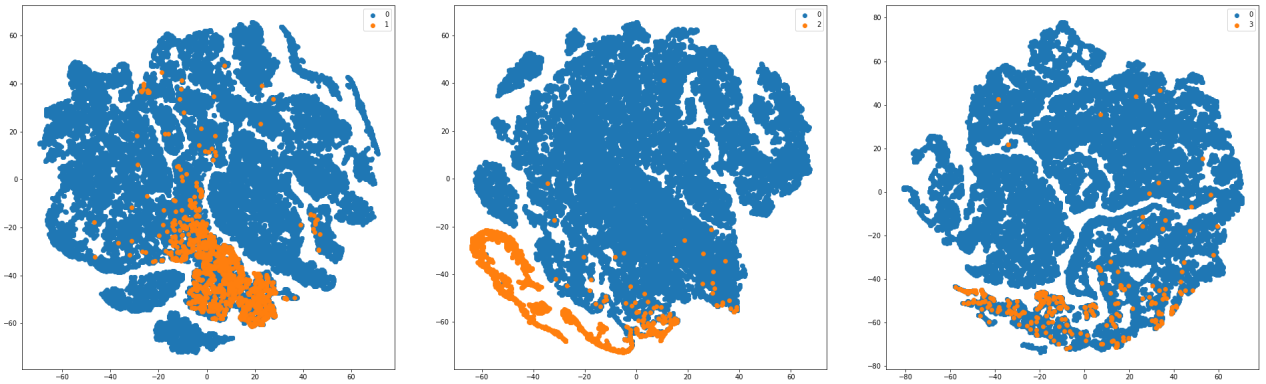
CNN-LSTM bagging models to classify VEB, SVEB and F classes individually using the one-vs-all scheme. The ratio 1:4 (i.e. the ratio of negative samples to positive samples is 4 to 1), gives the best results for all classes compared to other ratios of 1:2, 1:6 and 1:8. For SVEB class, the F1 score and the accuracy of 1:4 ratio are 30.04% and 84.31% that much higher than the other ratios. For VEB class, the accuracy and F1 score of all the ratios are generally high over 97% and 80% respectively. For F class, the ratios of 1:2, 1:4, and 1:8 have high accuracy at 95.72%, 92.97%, and 90.22% respectively. The F1 score of 1:4 ratio is 9.13% and it is higher than the F1 score of 1:2 ratio which is 8.62%.

The second experiment examines the effect of the number of consecutive heartbeats on a bag of CNN-LSTM and RRHOS-LSTM classifiers where each bag is combined separately. We train 10 binary CNN-LSTM bagging models on a different number of heartbeats: 1 beat, 3 beats, and 5 beats, where each model is trained individually on each class (SVEB, VEB, or F) using the one-vs-all scheme. The final decision is obtained by applying a majority voting per class on the bagging models. As shown in Table 2, the performance of CNN-LSTM classifiers is improved with the increasing number of input heartbeats. Using 5 beats as input achieves the highest performance for the classification of all classes. In F detection, for instance, the accuracy of CNN-LSTM using 5 beats is 92.97% compared to 87.12% and 65.65% for using 3 and 1 beats respectively. F1 score of 5 beats is 9.13%. It is much higher than 1 and 3 beats (4.11% and 5.95% respectively). Both 3 beats and 5 beats achieve comparable accuracy for the classification of VEB with 98.21% and 98.47% respectively. 5 beats F1 score is 88.87% higher than the 1 and 3 beats (51.48% and 87.37%). In SVEB detection, using 3 or 5 beats has a much higher sensitivity (84.27% and 81.63%) than using one beat (67.22%). F1 score for the 5 beats CNN-LSTM is 30.04% compared to 27.97% and 27.63% for CNN-LSTM based on 3 beats and 1 beat respectively.

Table 2 also shows in rows 4 and 5 the result of using a bag of RRHOS-LSTM models. HOS is tested to be extracted from 3 to 5 consecutive heartbeats and combined with RR intervals features which it is computed around the middle beat in the input sequence. RR intervals features are the same in both 3 and 5 beats based RRHOS-LSTM classifier.

The performance of a bag of RRHOS-LSTM classifiers based on 5 beats is better than 3 beats based. For instance, for the VEB class, the 5 beats based models resulted in an accuracy of 96.21%, a sensitivity of 94.62%, a specificity of 96.32%, and an F1 score of 76.41% compared to 96.07%, 94.31%, 96.20%, and 75.69% respectively for 3 beats based models. For the SVEB class, the RRHOS-LSTM based on 5 beats have higher accuracy and F1 score of 69.22% and 18.43% compared to 67.00% and 17.13%. For F class, positive predictivity and specificity of the 5 beats based models are better with 4.22% and 84.51% compared to 3.31% and 78.71% for the 3 beats based models. As notice, the CNN-LSTM model gives better performance than the RRHOS-LSTM model. However, the sensitivity of the RRHOS-LSTM is superior for all classes which will improve the overall performance of the combined model.

In the next experiment, a comparison between the fusion classifier and the majority voting (see Table 3) to combine the bagging models is performed. The fusion classifier (Table 3 row 1) aggregates the CNN-LSTM and RRHOS-LSTM models. The result of each individual bagging model is shown in Table 2 rows 3 and 5 respectively. The fusion classifier that combines both models demonstrates a significant improvement than the individual models. For example, For VEB class, F1 score, and the positive predictive value (93.47%, and 91.41%) are much higher than each individual bagging models of CNN-LSTM (88.87%, and 83.88%) and RRHOS-LSTM (76.41%, and 64.08%). For F class, the F1 score is 20.80% higher than the individual bagging models. For SVEB class, the fusion classifier achieves performance better than both bagging models. for instance, the F1 score of using



(a) (b) (c)

FIGURE 8. t-SNE visualization for the learned representation of the fusion network on the test data. (a) For SVEB class (orange color) and all other classes (blue color). (b) For VEB class (orange color) and all other classes (blue color). (c) For F class (orange color) and all other classes (blue color).

TABLE 4. Confusion matrix over (DS2) MIT-BIH of the proposed method using the fusion classifier and the verification network.

Reference	Predictions			
	N	SVEB	VEB	F
N	43,088	659	108	97
SVEB	689	1341	14	3
VEB	167	24	3020	5
F	259	2	52	75

the fusion classifier is 32.71% compared to 30.04% and 18.43% for each CNN-LSTM and RRHOS-LSTM bagging respectively.

The results of using the majority voting is shown in Table 3 (row 2). There is a noticeable enhancement by using the fusion classifier than the majority voting. The accuracy of the fusion classifier for SVEB and F classes (86.40% and 97.71%) is much higher than the majority voting (78.65% and 91.39%). Both methods have a comparable performance on the VEB class. In contrast to the majority voting, the SVEB detection is improved by using the fusion classifier.

Next, the verification network is employed to validate the fusion classifier results as shown in Table 3 (row 3). The verification network is a single deep neural network based on CNN-LSTM architecture. It examines the output of the fusion classifier to decide whether the positive classification label is correct or not. The verification network shows a significant improvement on the performance of the ECG arrhythmia classification system. For VEB class, the F1 score and positive productive value have improved to 94.23% and 94.55% compared to using the fusion classifier only (93.47% and 91.41%) respectively. For class SVEB, the F1 score and positive productive value (65.85% and 66.19%) are much better than the fusion classifier (32.71% and 20.55%) with much more specificity (98.56%) than the fusion classifier (86.67%). For the F class, the verification network is substantially increased the positive productively and F1 score of the proposed system by 27.41% and 5.61% respectively. The sensitivity of the verification network is lower than the fusion classifier for all, however the overall performance is much better. Table 4 shows the confusion matrix of the proposed method using the fusion classifier and the verification network. The normal class N is labeled if the testing sample is classified as negative by all binary fusion and verification classifiers.

In order to visualize the learned representation, we used the t-distributed stochastic neighbor embedding (t-SNE) method [41]. The t-SNE is a nonlinear dimension reduction method that mapping high-dimensional data into a space of two or three dimensions. The t-SNE is applied on the 500-dimensional vectors of the last fully connected layer (before the softmax layer) of the fusion classifier, as shown in Figure 8. The fusion classifier is trained on SVEB, VEB, and F classes separately as a binary classifier. Figure 8 (a) and (b) show a good clustering and localization for SVEB and VEB classes respectively. For the F class (Figure 8 (c)), the cluster is appearing however, some feature points are not localized enough.

Table 5 demonstrates the performance of the proposed method compared to varies state-of-the-art methods that are evaluated on the MIT-BIH DS2 dataset using the patient independent evaluation scheme. The proposed method is compared to the following methods: Sellami & Hwang [19], Shi et al [10], ensemble SVM [11], Raj & Ray [26], Mathews et al [25], Herry et al [28], Zhang et al [9], Ye et al [12], and Chazal et al [8]. We compute the five evaluation metrics for each class and add the average of sensitivity, specificity, positive predictive value, and F1 score. The overall accuracy is also computed. All these metrics are computed from the confusion matrix provided by the authors. The proposed method achieves superior performance compared to the-state-of-art heartbeats classification methods. The overall accuracy of the proposed method is 95.81% much higher than all the other methods. The second highest

TABLE 5. A comparison between the proposed method and the state-of-the-art techniques.

	N					SVEB					VEB				
	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	Acc
The proposed method	98.03	80.27	97.48	97.76	96.01	65.51	98.56	66.19	65.85	97.20	93.91	99.62	94.55	94.23	99.25
Sellami & Hwang [19]	88.52	91.31	98.81	93.38	88.82	82.04	92.80	30.44	44.41	92.40	92.05	97.55	72.22	80.93	97.19
Shi et al [10]	91.90	95.83	99.45	95.53	92.33	91.67	95.68	44.86	60.24	95.54	95.12	99.11	88.09	91.47	98.85
Ensemble SVM [11]	95.94	86.34	98.20	97.06	94.84	78.10	96.61	49.75	60.78	95.84	94.75	99.57	93.79	94.27	99.25
Raj & Ray [26]	91.04	95.33	99.38	95.03	91.51	80.93	96.73	48.75	60.85	96.14	84.47	99.03	85.42	84.94	98.10
Mathews et al [25]	74.64	97.80	99.66	85.35	77.08	88.50	93.26	33.64	48.75	93.08	77.75	97.79	69.09	73.16	96.59
Herry et al [28]	83.13	92.71	98.93	90.35	84.18	81.14	93.36	31.93	45.83	92.90	77.50	98.58	79.05	78.27	97.22
Zhang et al [9]	88.94	92.84	98.98	93.69	89.38	79.06	93.95	35.98	49.46	93.33	85.48	99.54	92.75	88.96	98.63
Ye et al [12]	88.61	82.28	97.55	92.87	87.91	61.02	97.71	52.34	56.34	96.27	81.82	96.46	61.45	70.19	95.52
Chazal et al [8]	87.06	93.95	99.17	92.72	87.80	75.98	95.33	38.53	51.13	94.61	80.31	98.79	81.67	80.98	97.62

	F					Average					
	Se	Sp	PPv	F1	Acc	Se	Sp	PPv	F1	κ	Acc
The proposed method	19.33	99.79	41.67	26.41	99.16	69.20	94.56	74.97	71.06	**0.79**	**95.81**
Sellami & Hwang [19]	68.30	98.51	26.58	38.27	98.28	82.72	95.04	57.01	64.25	0.58	88.35
Shi et al [10]	61.60	97.29	15.16	24.33	97.01	85.07	96.98	61.89	67.89	0.68	91.87
Ensemble SVM [11]	12.37	99.69	23.65	16.24	99.00	70.29	95.55	66.35	67.09	0.75	94.47
Raj & Ray [26]	93.56	94.80	12.44	21.96	94.79	87.50	96.47	61.50	65.69	0.63	90.27
Mathews et al [25]	93.75	84.17	4.68	8.91	84.25	83.66	93.26	51.77	54.04	0.37	75.50
Herry et al [28]	83.25	91.15	6.91	12.75	91.09	81.26	93.95	54.21	56.80	0.46	82.70
Zhang et al [9]	93.81	95.36	13.73	23.96	95.35	86.82	95.42	60.36	64.02	0.59	88.34
Ye et al [12]	19.69	93.99	2.50	4.43	93.41	62.79	92.61	53.46	55.96	0.51	86.55
Chazal et al [8]	89.43	92.46	8.57	15.64	92.44	83.19	95.13	56.98	60.12	0.53	86.24

TABLE 6. Comparing the training time (seconds), testing time (seconds), and the number of parameters of the CNN-LSTM, RRHOS-LSTM, the fusion classifier and the verification network in seconds.

	training time	testing time	Parameters
CNN-LSTM	18.67	-	$3M$
RRHOS-LSTM	3.5	-	$54K$
Fusion classifier	15	0.0004	$272K$
Ver. Net	22	0.0003	$3M$

method is the ensemble SVM [11] with an overall accuracy of 94.47%. The proposed method has a much higher Cohen's Kappa coefficient (0.79) compared to the next best method (0.75). This shows there is substantial agreement between the proposed method and the ground truth. The proposed method has the highest average F1 score and average positive predictive value (71.06% and 74.97%) among all the other methods with more than 8% and 3% improvement compared to the next highest method. The proposed method has the highest positive predictive value for classes SVEB, VEB, and F. This means that our method is more likely to truly recognize the abnormal classes than the other methods. For the specificity, the proposed method achieves superior values comparing to the state-of-the-art methods for the classes, VEB, SVEB, and F (99.62%, 98.56%, and 99.79%). For the sensitivity, the proposed method has the highest value for class N (98.03%) and ranked the third for VEB class (93.91%) compared to the-state-of-art methods. The sensitivity of classes F and SVEB (19.33% and 65.51%) are not among the highest values due to the verification network reduces the true positive values of these classes in exchange for a significant reduction in the false-positive values.

Table 6 shows the training and the testing time of the different proposed models. The training times of a single CNN-LSTM and RRHOS-LSTM are 18.67 and 3.5 seconds respectively. For each ECG arrhythmia class, we train 20 bagging models (i.e. 10 for CNN-LSTM and 10 for RRHOS-LSTM) so that the total training time is 11.1 minutes. After that, we train the fusion classifier to combine 20 models into a single classifier. It takes 15 seconds to train one fusion classifier. The verification network needs 22 seconds to be trained. The total training time of the different proposed models over all classes is 13 minutes. The inference times of the fusion classifier and the verification network for a single beat are 0.4 and 0.3 milliseconds respectively. Note, the inference time of the fusion classifier also includes the inference time of the CNN-LSTM and RRHOS-LSTM bagging models.

V. CONCLUSION

In this paper, a novel deep learning-based multi-model ensemble is proposed which achieves superior classification performance compared to the-state-of-the-art methods. The proposed multi-model system consists of two different deep learning bagging models. The first model is based on the CNN and LSTM architectures and takes the raw ECG beats as an input. The second model is based on a combination of classical feature, i.e. RR intervals and HOS, and LSTM model. Each model is trained on a sub-sample of the training set using the bagging scheme. The deep learning bagging models are fused using a meta-classifier. The result of the fusion classifier is refined using another deep learning network to verify the abnormal classes and reduce the false positive.

The proposed method is evaluated on the standard MIT-BIH dataset to classify the heartbeats into four classes according to AAMI recommendation using the ''subject-oriented'' patient independent scheme. The overall accuracy of the proposed method is the highest (95.81%) by more than 1% from the nearest state-of-the-art method. The proposed method provided positive predictive values of 97.48%, 66.19%, 94.55%, and 41.67% for N, SVEB, VEB, and F classes. The sensitivities for these four classes were 98.03%, 65.51%, 93.91%, and 19.33%. The averages of positive predictive value and F1 score across all classes are higher than all other methods by more than 8% and 3% respectively. Results showed that the proposed ensemble of multi-model deep learning can pick up useful information from using multiple ECG beats as an input. The ensemble of different deep learning models using the meta-learner approach and then verifying it, allowing the proposed system to outperform state-of-the-art arrhythmia classifiers that based on either traditional machine learning methods or deep neural networks.

REFERENCES

[1] H. V. Huikuri, A. Castellanos, and R. J. Myerburg, ''Sudden death due to cardiac arrhythmias,'' *New England J. Med.*, vol. 345, no. 20, pp. 1473–1482, 2001.

[2] Z. Binici, T. Intzilakis, O. W. Nielsen, L. Køber, and A. Sajadieh, ''Excessive supraventricular ectopic activity and increased risk of atrial fibrillation and stroke,'' *Circulation*, vol. 121, no. 17, pp. 1904–1911, May 2010.

[3] U. Ofoma, F. He, M. L. Shaffer, G. V. Naccarelli, and D. Liao, ''Premature cardiac contractions and risk of incident ischemic stroke,'' *J. Amer. Heart Assoc.*, vol. 1, no. 5, Sep. 2012, Art. no. e002519.

[4] (May 2017). *Cardiovascular Diseases (CVDs)* [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-disease%s-(cvds)

[5] L. Biel, O. Pettersson, L. Philipson, and P. Wide, ''ECG analysis: A new approach in human identification,'' *IEEE Trans. Instrum. Meas.*, vol. 50, no. 3, pp. 808–812, Jun. 2001.

[6] G. J. Taylor, *150 Practice ECGs: Interpretation and Review*. Hoboken, NJ, USA: Wiley, 2006.

[7] *Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms*, document ANSI/AAMI EC57, Association for the Advancement of Medical Instrumentation, 1998.

[8] P. de Chazal, M. O'Dwyer, and R. B. Reilly, ''Automatic classification of heartbeats using ECG morphology and heartbeat interval features,'' *IEEE Trans. Biomed. Eng.*, vol. 51, no. 7, pp. 1196–1206, Jul. 2004.

[9] Z. Zhang, J. Dong, X. Luo, K.-S. Choi, and X. Wu, ''Heartbeat classification using disease-specific feature selection,'' *Comput. Biol. Med.*, vol. 46, pp. 79–89, Mar. 2014.

[10] H. Shi, H. Wang, Y. Huang, L. Zhao, C. Qin, and C. Liu, ''A hierarchical method based on weighted extreme gradient boosting in ECG heartbeat classification,'' *Comput. Methods Programs Biomed.*, vol. 171, pp. 1–10, Apr. 2019.

[11] V. Mondéjar-Guerra, J. Novo, J. Rouco, M. G. Penedo, and M. Ortega, ''Heartbeat classification fusing temporal and morphological information of ECGs via ensemble of classifiers,'' *Biomed. Signal Process. Control*, vol. 47, pp. 41–48, Jan. 2019.

[12] C. Ye, B. V. K. V. Kumar, and M. T. Coimbra, ''Heartbeat classification using morphological and dynamic features of ECG signals,'' *IEEE Trans. Biomed. Eng.*, vol. 59, no. 10, pp. 2930–2941, Oct. 2012.

[13] T. Ince, S. Kiranyaz, and M. Gabbouj, ''A generic and robust system for automated patient-specific classification of ECG signals,'' *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1415–1426, May 2009.

[14] S. Osowski, L. T. Hoai, and T. Markiewicz, ''Support vector machine-based expert system for reliable heartbeat recognition,'' *IEEE Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 582–589, Apr. 2004.

[15] G. D. Lannoy, D. François, J. Delbeke, and M. Verleysen, ''Weighted SVMs and feature relevance assessment in supervised heart beat classification,'' in *Biomedical Engineering Systems and Technologies*. Berlin, Germany: Springer, 2011, pp. 212–223.

[16] K. N. V. P. S. Rajesh and R. Dhuli, ''Classification of imbalanced ECG beats using re-sampling techniques and AdaBoost ensemble classifier,'' *Biomed. Signal Process. Control*, vol. 41, pp. 242–254, Mar. 2018.

[17] S. Kiranyaz, T. Ince, and M. Gabbouj, ''Real-time patient-specific ecg classification by 1-D convolutional neural networks,'' *IEEE Trans. Biomed. Eng.*, vol. 63, no. 3, pp. 664–675, Mar. 2016.

[18] U. R. Acharya, H. Fujita, S. L. Oh, Y. Hagiwara, J. H. Tan, and M. Adam, ''Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network,'' *Inf. Sci.*, vol. 405, no. 1, pp. 81–90, Sep. 2017.

[19] A. Sellami and H. Hwang, ''A robust deep convolutional neural network with batch-weighted loss for heartbeat classification,'' *Expert Syst. Appl.*, vol. 122, pp. 75–84, May 2019.

[20] S. L. Oh, E. Y. K. Ng, R. S. Tan, and U. R. Acharya, ''Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats,'' *Comput. Biol. Med.*, vol. 102, no. 1, pp. 278–287, Nov. 2018.

[21] B. Hou, J. Yang, P. Wang, and R. Yan, ''LSTM-based auto-encoder model for ECG arrhythmias classification,'' *IEEE Trans. Instrum. Meas.*, vol. 69, no. 4, pp. 1232–1240, Apr. 2020.

[22] O. Yildirim, U. B. Baloglu, R.-S. Tan, E. J. Ciaccio, and U. R. Acharya, ''A new approach for arrhythmia classification using deep coded features and LSTM networks,'' *Comput. Methods Programs Biomed.*, vol. 176, pp. 121–133, Jul. 2019.

[23] S. S. Xu, M.-W. Mak, and C.-C. Cheung, ''Towards end-to-end ECG classification with raw signal extraction and deep neural networks,'' *IEEE J. Biomed. Health Inform.*, vol. 23, no. 4, pp. 1574–1584, Jul. 2019.

[24] S. Saadatnejad, M. Oveisi, and M. Hashemi, ''LSTM-based ECG classification for continuous monitoring on personal wearable devices,'' *IEEE J. Biomed. Health Informat.*, vol. 24, no. 2, pp. 515–523, Feb. 2020.

[25] S. M. Mathews, C. Kambhamettu, and K. E. Barner, ''A novel application of deep learning for single-lead ECG classification,'' *Comput. Biol. Med.*, vol. 99, pp. 53–62, Aug. 2018.

[26] S. Raj and K. C. Ray, ''Sparse representation of ECG signals for automated recognition of cardiac arrhythmias,'' *Expert Syst. Appl.*, vol. 105, pp. 49–64, Sep. 2018.

[27] Ö. 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.

[28] C. L. Herry, M. Frasch, A. J. E. Seely, and H.-T. Wu, ''Heart beat classification from single-lead ECG using the synchrosqueezing transform,'' *Physiol. Meas.*, vol. 38, no. 2, pp. 171–187, Feb. 2017.

[29] J. Jiang, H. Zhang, D. Pi, and C. Dai, ''A novel multi-module neural network system for imbalanced heartbeats classification,'' *Expert Syst. Appl., X*, vol. 1, Apr. 2019, Art. no. 100003.

[30] J. Pan and W. J. Tompkins, ''A real-time QRS detection algorithm,'' *IEEE Trans. Biomed. Eng.*, vol. BME-32, no. 3, pp. 230–236, Mar. 1985.

[31] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[32] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, ''Inception-v4, Inception-ResNet and the impact of residual connections on learning,'' in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4278–4284.

[33] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin, ''Convolutional sequence to sequence learning,'' in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70. Sydney, NSW, Australia: International Convention Centre, Aug. 2017, pp. 1243–1252.

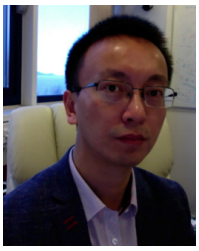
[34] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, ''A convolutional neural network for modelling sentences,'' in *Proc. 52nd Annu. Meeting Assoc. Comput. Comput. Linguistics*, vol. 1, Jun. 2014, pp. 655–665.

[35] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, ''Convolutional neural networks for speech recognition,'' *IEEE/ACM Trans. Audio, Speech Language Process.*, vol. 22, no. 10, pp. 1533–1545, Oct. 2015.

[36] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, ''A novel connectionist system for unconstrained handwriting recognition,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 5, pp. 855–868, May 2009.

[37] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Sep. 2015, pp. 1412–1421.
[38] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng. Med. Biol. Mag.*, vol. 20, no. 3, pp. 45–50, May/Jun. 2001.
[39] 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.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
[40] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, and M. Kudlur, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Conf. Operating Syst. Design Implement.*, 2016, pp. 265–283.
[41] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

EHAB ESSA received the B.Sc. and M.S. degrees from Mansoura University, Egypt, in 2004 and 2008, respectively, and the Ph.D. degree from Swansea University, U.K., in 2014, all in computer science. He was a COFUND Fellow with the Department of Computer Science, Swansea University. He is currently an Associate Professor with the Department of Computer Science, Mansoura University. He is the author of over 32 peer-reviewed articles. His current research interests include deep learning, medical image analysis, the IoT, computer vision, and artificial intelligence.



XIANGHUA XIE (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in computer science from the University of Bristol, Bristol, U.K., in 2002 and 2006, respectively. He is currently a Professor with the Department of Computer Science, Swansea University, Swansea, U.K., where he is also leading the Computer Vision and Machine Learning Laboratory. His research interests include various aspects of pattern recognition and machine intelligence and their applications to real-world problems. He has authored or coauthored more than 160 refereed conference and journal publications and coedited several conference proceedings. He is a member of BMVA. He was a recipient of the RCUK Academic Fellowship. He is also an Associate Editor of several journals, including *IET Computer Vision*.