International Journal of Science and Research Archive

eISSN: 2582-8185

Cross Ref DOI: 10.30574/ijsra

Journal homepage: https://ijsra.net/

IJSRA

(Review Article)

Integrating deep learning architectures for improved arrhythmia detection in electrocardiogram signals

Hima Vijayan VP *, Reshma S, R. Sathyanjan and Vaishnavi Geetha

Department of ECE, ACE College of Engineering & APJ Abdul Kalam Technological University Thiruvananthapuram, India.

International Journal of Science and Research Archive, 2023, 08(02), 704–715

Publication history: Received on 17 March 2023; revised on 29 April 2023; accepted on 01 May 2023

Article DOI: https://doi.org/10.30574/ijsra.2023.8.2.0332

Abstract

Cardiovascular ailment (CVD)is a main reason of morbidity and mortality worldwide. WHO reviews that there are almost 1 crore 20 Lakhs of deaths due to coronary heart illnesses. Early detection and correct prognosis of CVDs are important for powerful remedy and prevention. Monitoring the affected person for twenty-four hrs isn't continually feasible as it calls for plenty of understanding and time. Heart disorder remedy or prognosis are very complicated, specifically in growing countries or poor nations. The clinical enterprise has a big quantity of statistics and is constantly utilized by researchers to increase advancing technology and generation to limit sizeable quantity of deaths because of coronary heart sicknesses. A good buy of statistics mining and ML strategies or algorithms are to be had to fetch the statistics from databases and use this fetched statistics to expect the coronary heart sicknesses very accurately. This paper offers a comparative study of those 3 deep learning architectures for arrhythmia detection using electrocardiogram (ECG) signals. The overall performance of the models become evaluated using standard metrics together with accuracy, sensitivity, and specificity.

Keywords: Congenital heart disease; Recurrent Neural Network; Rectified Linear Network; Cardiac Arrhythmia

1. Introduction

Cardiovascular disease is a term that refers broadly to a set of conditions that may affect our heart and blood vessels. CVDs are one of the major diseases threatening human life that can also lead to death.WHO reports that, among all causes of death in today's world, cardiovascular diseases ranks first. If left untreated,CVDs can have devastating consequences.So to prevent CVDs, it has become important to regularly monitor the rhythm of the most vital part of human body, the heart.

As we all know, a healthy heart is essential for over well-being and overall health. Regular monitoring of heart can help detect problems at early stages. But today, early detection and treatment of CVDs are uncommon. The first and foremost step is to detect any kind of cardiovascular disease early.For the past few years, many deep learning algorithms and techniques have been proposed for early detection of heart disease. Some of them are Convolutional Neural Network(CNN), Long short-term memory(LSTM), Visual Geometry Group(VGG).

This paper tests the CNN machine learning model for detection of CVD and adds more convolutional layers to improve the accuracy of the model. A comparative study on the accuracy of this machine learning models is also being performed.

As a novel addition, this paper integrated a new feature i.e. Arrhythmia is further classified into tachycardia and bradycardia and the model was deployed and it demonstrated effective results.

* Corresponding author: Hima Vijayan VP

2. Literature survey

Kuldeep Vayadande. et.al.[1] in this heart disease model, machine learning and deep learning algorithms are used and all algorithms are implemented in the dataset. The dataset used comes from Kaggle, consisting of 303 rows and 14 attributes. The algorithms used in the model are Logistic Regression, NB, K-NN, SVM, Random Forest, Multilayer Perceptron, , XG Boost, Artificial Neural Network, Decision Tree and Cat Boost. The accuracy obtained from logistic regression was 88.52%. , Random Forest is 88.52% and according to XGBoost is 88.52 %. Deep learning techniques like Multilayer Perceptron and Artificial Neural Network are used in this model, the performance is quite good. Accuracy for Multilayer Perceptron is 86.89% and Artificial Neural Network is 85.25%.

D.P.Yadav et al.[2] in this study, an automatic heart disease prediction system was developed using machine learning techniques and feature optimization to help doctors. The UCI dataset contains 14 attributes. These attributes were used to train and classify SVM, KNN, Naïve Bayes and Random Forest. Among these models, the naive basis achieved the highest accuracy of 87.9% using triple cross-validation. A GA feature optimization technique was applied. Triple cross-validation is applied to the data set so that unbiased performance can be measured. The base Naïve model is 96% accurate.

R.Deepika.. et.al.[3] introduces deep learning techniques used to predict cardiac abnormalities. Deep learning techniques help process image data through various tests including electrocardiogram, echocardiogram, etc. to identify cardiac anomalies. This template is used to periodically monitor patient diagnostic data for rapid detection. The main objective of this paper is to improve the performance of the prediction system by detecting all the features associated with cardiac anomalies.

Mohammed.M.Farag et al[4] presented the automated electrocardiogram (ECG) classification for arrhythmia monitoring that is central to the diagnosis of cardiovascular disease. Machine learning (ML) is widely used to detect arrhythmias. Cloud-based inference is the dominant implementation model of modern ML algorithms that do not always meet the availability and privacy requirements of ECG monitoring. Edge inference is an emerging alternative that solves latency, privacy, connectivity, and availability issues. However, implementing ML models at the edge is challenging due to the high requirements of modern ML algorithms and the computational limitations of edge devices. In this work, a short-term, lightweight, standalone Fourier transform convolutional neural network (STFT) model for real-time ECG classification and edge arrhythmia detection is proposed. An explicit interpretation of the convolutional layer as a Finite Layer Impulse Response Filter (FIR) and exploiting this interpretation to develop a STFT-based 1D convolutional layer (Conv1D) to extract the spectrum from the input ECG signal entry is also provided.

Yun Kwan Kim.. et.al [5] In this study in , a new automatic classification framework combining residual network with compression and excitation block as well as two- dimensional long-term memory is proposed. Performance at levels eight, four, and two was evaluated on the MIT-BIH Arrhythmia Database (MITDB), the MIT-BIH Atrial Fibrillation Database (AFDB), and the PhysioNet/Internal Calculator. cardiology challenge 2017 (CinC DB), and they outperformed the performance obtained by conventional methods. To measure the generalizability of the proposed framework, AFDB and CinC DB were tested using an MITDB-trained model and achieved superior performance over ShallowConvNet and DeepConvNet. Finally, the recommendation framework achieves the best generalization ability compared to other deep learning models. Therefore, we confirm that ResNet combined with SE block and biLSTM architecture classifies arrhythmias from single-lead ECG signals without feature extraction.

3. Electrocardiogram (ECG/EKG)

An electrocardiogram (ECG / EKG)[6] is a test that records the electrical activity of the heart. An electrocardiogram is a painless, noninvasive test that provides valuable information about the heart's functioning. An electrocardiogram involves placing small sensors, called electrodes, on the skin of the chest, arms, and legs. When the heart beats, the electrical signals detected by the electrodes are transmitted to the data logger.

An electrocardiogram is basically an electrical map of the heart. This is a voltage-over-time graph for the heart's electrical effort. Electrocardiograms are commonly used to detect and diagnose problems such as abnormal heart rhythms, heart muscle damage, and coronary artery disease [7].

The three main components in an ECG wave are the P, QRS and T waves[8].

P wave- indicating the contraction- depolarization of the upper chambers of the heart, the Atria.

QRS complex- shows the contraction- depolarization of lower chambers of the heart, the ventricles. QRS complex involves Q,R,S waves.

T wave- indicates the relaxation- repolarization of the ventricle

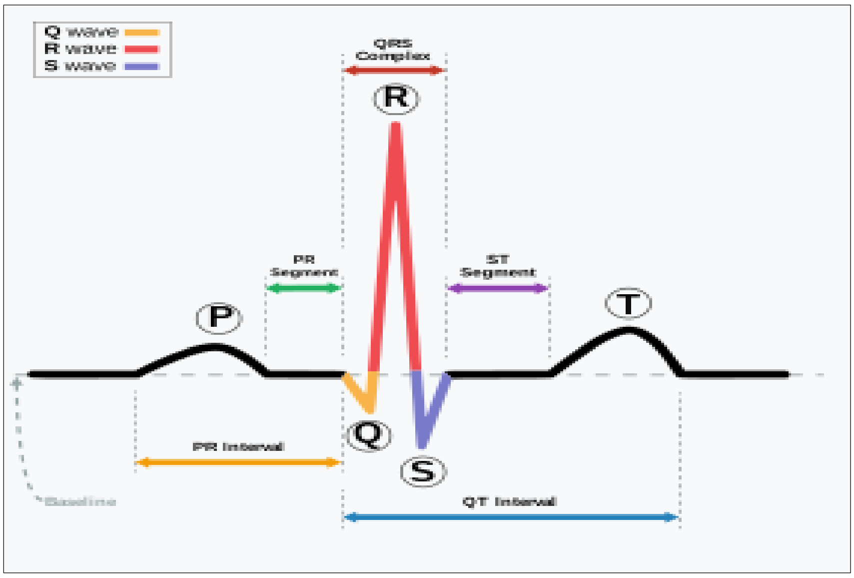

Figure 1 ECG Waves [9]

In a healthy heart, depolarization [10] during each heartbeat occurs in a precise sequence starting at the sinoatrial node and passing through the atria, atrioventricular node, bundle of His fibers, and purkinje[11]. , eventually spreading into the ventricles in a descending and contralateral direction. This sorted depolarization pattern results in a characteristic ECG trace.

The final ECG interpretation relies on pattern recognition. The theory is rooted in electromagnetism and can be distilled to the following points.

- Positive deflection produced by depolarization of heart towards positive electrode
- Negative deflection produced by depolarization of heart away from positive electrode
- Negative deflection produced by repolarization of heart towards positive electrode
- Positive deflection produced by repolarization of heart away from positive electrode

4. Arrhythmia

An arrhythmia, also known as an arrhythmia [12], or an ECG arrhythmia is a medical condition that refers to an irregular heartbeat. Sometimes the heart can beat too fast, too slow, or irregularly, leading to an arrhythmia.

It is normal for the heart rate to increase rapidly during any physical activity, to slow down during sleep or at rest, indeed it is normal to feel like the heart skips a beat from time to time. But arrhythmia is a common condition when this irregular heartbeat occurs. This means that the heart is not pumping enough blood to the body, which can lead to symptoms such as dizziness, fainting, and others.

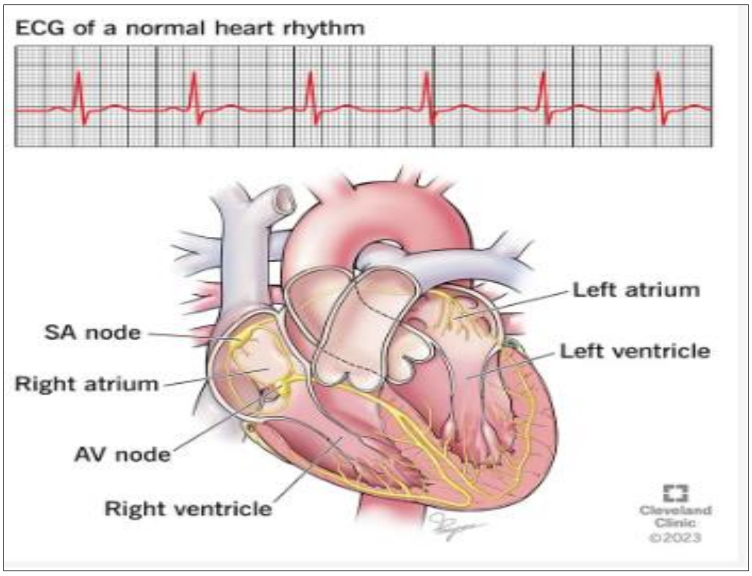

Figure 2 An Arrhythmia disrupts the way heartbeat signals normally travel through your heart.[13]

In some cases, arrhythmias will not have any symptoms. This is why it is so difficult to be detected early. Although most cases of arrhythmia are not serious, some can lead to problems such as stroke or heart failure. In the worst case, it can even lead to sudden death.

Causes of arrhythmias can vary and may include heart disease, congenital heart defects[14], electrolyte imbalances, medications, or other medical conditions. Some common causes of arrhythmias include high blood pressure, diabetes, hyperthyroidism, sleep apnea, drug abuse, family history of arrhythmias, or sudden cardiac arrest.

Based on different criteria, arrhythmias can be classified in several ways. Here are some common classifications.

By location in the heart:

- Atrial arrhythmias, in the upper chambers(atria) of the heart.[15]
- Ventricular arrhythmias, in the lower chambers(ventricles) of the heart.[16]

By heart rate:

- Bradycardia, heartrate below 60 beats per minute.[17]
- Tachycardia, heartbeats faster than 100 beats per minute.[18]
- Normal sinus rhythm, a regular heart rate between 60 and 100 beats per minute.

By pattern:

- Regular : the heart beats with a consistent pattern.
- Irregular : the heart beats inconsistently..

By severity:

- Asymptomatic, the arrhythmia is present but does not cause any symptoms or adverse effects.
- Symptomatic, the arrhythmia causes noticeable symptoms, such as palpitations, dizziness or chest pain.
- Life threatening, the arrhythmia can lead to serious complications, such as sudden cardiac arrest or stroke.

By Cause:

- Congenital arrhythmias[19],caused by structural abnormalities or genetic mutations present at birth.
- Acquired arrhythmias,caused by external factors, such as heart disease, medications or electrolyte imbalances.

5. Convolutional neural network

A convolutional neural network (CNN) [20] is a type of neural network commonly used in computer vision applications such as image and video recognition. It's a deep learning process that uses input images, notices different things/objects in the image and distinguishes them from others. The network consists of one or more layers that normally apply a set of filters or kernels[21] to the input data and the grouping layers, which reduces the spatial size of the output of the convolutional layers.

CNN can successfully capture spatial and temporal dependencies in input data through the application of relevant filters. This model is more suitable for image datasets due to reduced parameters and reusability of weights. In other words, the network can be trained to better understand the complexity of the input.

Convolutional layers are designed to learn features from input data, such as edges, lines, and shapes, by dragging filters over the input and performing multiplication and sum operations on each. Element [37]. Pooling layers are used to reduce the size of the output of the convolutional layers, which reduces the computation required by the network.

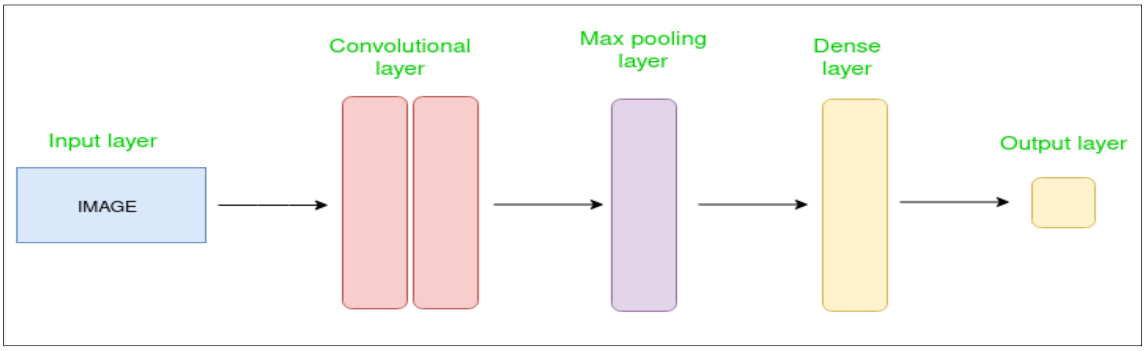

Figure 3 Simple CNN architecture[22]

CNNs are composed of multiple layers that work together to learn increasingly complex representations of the input data. These layers are typically organized into three layers: Convolutional layers, Pooling layers and fully connected layers.[23]

A convolutional layer is the core building block of a CNN. It consists of filter layers or kernels whose parameters will be learned during the training process. Filters are usually smaller than the actual image. For convolution the filter slid across the height and width of the image and the dot product between every element of the filter and the input is calculated at every spatial position. The output of this operation is a feature map that highlights the presence of certain pattern or features in the input data.

Similar to the convolutional layer, the Pooling layer is responsible for reducing the size of the convolutional features.This is to reduce the computational power required to process the data by reducing the size.It also helps to remove important features that are not compatible with rotation and position, thus controlling the training process of good models.

There are two types of pooling: max pooling and average pooling. Max Pooling returns the maximum value from a portion of the image covered by the kernel. On the other hand, Average pooling returns the average of all the values in the portion of the image covered by the kernel.

In addition to these layers, CNNs may also include other types of layers, such as dropout layers to prevent overfitting, and batch normalization layers to improved training stability.

During training, the network adjusts the values of the filters and biases in the convolutional layers, as well as the weights and biases in the fully connected layers to minimize the difference between predicted outputs and actual outputs. This process is known as back propagation and is typically implemented using stochastic gradient descent(SGD)[24] or a variant theorem.

6. Long short-term memory

Long short-term memory(LSTM)[25] is a Convolutional Neural Network used in artificial intelligence and deep learning. The peculiarity of LSTM is that it has feedback connections. It can be the entire data array, not just individual content.

An RNN[26] model has both long and short-term memory. The LSTM architecture aims to provide a short term memory for RNN that can last thousands of timesteps, thus long " short-term memory".

LSTM networks are designed to address the problem of vanishing and exploding gradients in traditional RNNs. They have additional memory cells known as the "gates" which allow the network to selectively store and forget information, making them particularly effective for processing sequential data.

Traditional RNNs have a simple structure where each hidden state at a given time step is a function of the current input and the previous hidden state. But the gradients used for updating network's parameters become either too small or too large as they are propagated backwards through time. This can lead to slow convergence and poor performance.

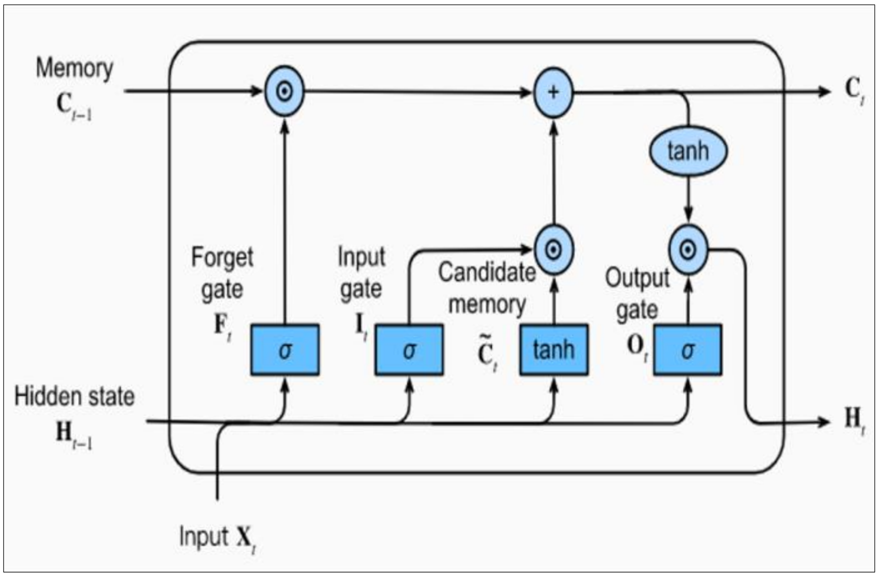

Figure 4 LSTM Architecture[27]

LSTM have gates which are composed of sigmoid and element wise multiplication operations that decide what information to keep or forget from the previous hidden state and what information to update with the current input.

The three gates[28] in LSTM are:

- Forget gate: This gate determines which information from the previous hidden state should be forgotten. It takes the previous hidden state and the current input as input and produces a vector of values between 0 and 1 for each element in the hidden state. Values close to 0 indicate that the corresponding information should be forgotten, while values close to 1 indicate that it should be kept.
- Input gate: This gate determines which new information from the current input should be added to the memory cell. It takes the previous hidden state and the current input as input and produces a vector of values between 0 and 1 for each element in the hidden state. Values close to 0 indicate that the corresponding information should not be added, while values close to 1 indicate that it should be added.
- Output gate: This gate determines which information from the memory cell should be output. It takes the previous hidden state and the current input as input, as well as the current memory cell state, and produces a vector of values between 0 and 1 for each element in the hidden state. Values close to 0 indicate that the corresponding information should not be the output while values close to 1 indicate that it should be the output.

LSTM networks are well-suited for classification, processing and forecasting based on temporal data, where they may be long-term trade-offs between significant events over time. The relative insensitivity to variable length is an advantage of LSTMs over RNNs, latent Markov models[29], and other sequential learning methods in many applications.

7. Visual geometry group-16

VGG-16 [30] is a deep learning architecture that uses convolutional neural networks (CNNs) to recognize and classify images. The architecture is named VGG because it was developed by the Visual Geometry Group at the University of Oxford, and the number 16 refers to the number of layers in the network.

The VGG-16 architecture has 13 convolutional layers and three fully connected layers. The convolutional layers are grouped into 5 blocks, where each block contains multiple convolutional layers followed by a max pooling layer. The convolutional layers in each block have a fixed number of filters, which is either 64 or 128. The max pooling layers have a fixed pool size of 2x2 and a stride of 2, which reduces the spatial dimensions of the feature maps by half. Each convolutional layer applies a set of filters to the input image to extract features such as edges, textures, and patterns. The filters are learned during the training process, and the output of each convolutional layer is passed through a non-linear activation function such as ReLU (rectified linear unit)[31].

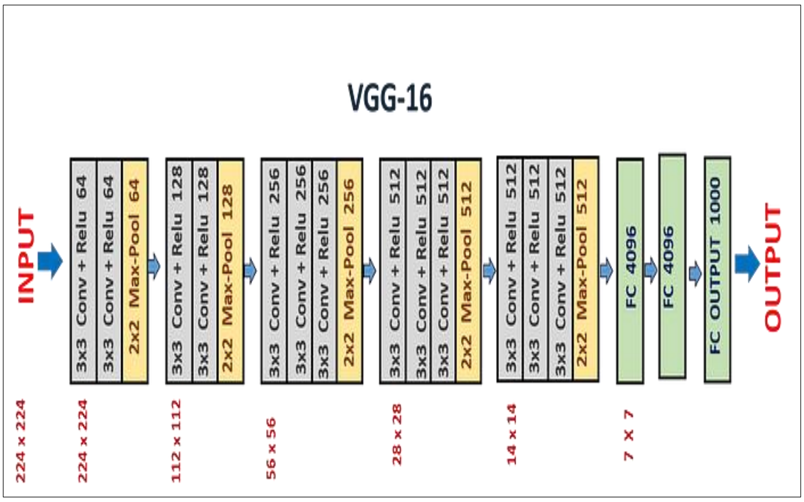

Figure 5 VGG-16 Architecture[32]

The input image is first normalized to have zero mean and unit variance. The normalized image is then passed through the convolutional layers to extract features. Each convolutional layer applies a set of filters to the input feature map to detect specific patterns in the image, such as edges, corners, and textures. The filters are learned during the training process by backpropagating the error gradients.

The output of the last convolutional layer is a feature map with spatial dimensions of 7x7 and 512 channels. This feature map is then flattened and passed through the fully connected layers to perform the final classification task. The first two fully connected layers have 4,096 neurons each, while the last fully connected layer has 1,000 neurons, corresponding to the number of classes in the ImageNet[33] dataset. The output of the last fully connected layer is passed through a softmax activation function to obtain the predicted probability distribution over the classes.

VGG-16 is known for its simplicity and effectiveness.The network is widely used as a feature extractor for various computer vision tasks such as object detection, segmentation, and image retrieval. One of the key features of the VGG-16 architecture is its simplicity and uniformity. All the convolutional layers have the same filter size of 3x3, and all the max pooling layers have the same pool size and stride. This simplicity makes the architecture easy to understand and implement, and it also helps to avoid overfitting. The uniformity of the architecture also makes it easy to transfer the learned features to other tasks, such as object detection and segmentation, by replacing the last fully connected layer with a new layer that is specific to the task.

8. Implementation

8.1. Data collection

- Accurately predicting arrhythmia benefits from obtaining lots of data.
- Both patients with and without arrhythmia are included in our data.
- A dataset that is frequently employed in studies on arrhythmia identification using machine learning techniques is the MIT-BIH Arrhythmia Database[34]. It has 48 ECG recordings spanning half an hour, captured at 360 Hz, with a total of 23 different forms of arrhythmia and normal rhythm noted by cardiologists. 47 patients, varying in age and condition, including healthy individuals and those with different types of heart disease, provided the recordings. This dataset has long been used to research and forecast arrhythmia.
- Binary and ECG annotations are the two accessible formats.

8.2. Import the necessary libraries

- Starting with "wfdb", which is the primary library for PhysioNet data work, including the MIT-BIH Arrhythmia Database. It offers tools for reading, creating, and modifying annotations and ECG signals.
- Python's "numpy" package is a foundational tool for performing scientific computations. In addition to functions for numerical operations, it offers support for multi-dimensional arrays and matrices[35].
- Python's visualisation library is called "matplotlib". It offers tools for making plots and charts that can be used to illustrate descriptions and ECG signals.
- "Pandas" is a data analysis and manipulation library. In addition to features for data processing and aggregation, it offers support for reading and writing data in a number of formats.
- A popular machine learning library for Python is "sklearn". It provides functions for data pre-processing, model selection, and evaluation, as well as a wide range of machine learning algorithms.
- "TensorFlow"[36] is a popular deep learning library for Python that can be used for arrhythmia detection using neural networks. Once you have imported TensorFlow, you can use it to define, train, and evaluate various types of neural networks, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for arrhythmia detection. TensorFlow provides a wide range of functions and classes for building and training neural networks, as well as tools for data pre-processing, model visualization, and deployment. It also supports distributed computing and can run on multiple GPUs or CPUs to accelerate training and inference.

8.3. Load and pre-process data

- Extraction of the .dat and.atr files for the recordings and annotations should be done after downloading the MIT-BIH Arrhythmia Database from PhysioNet.
- Utilise the wfdb package to load the ECG recordings and annotations into Python.
- Filtering, resampling, and normalising the ECG signals as needed are all examples of pre-processing. This may entail scaling the signals to a common range, applying bandpass filters to reduce noise and artefacts, and resampling the signals at a constant sampling rate.
- Use the annotations to divide the pre-processed ECG signals into distinct beats or segments. The R-peaks from the annotations may then be extracted and used as segmentation reference points.
- Using the annotations, assign the corresponding arrhythmia kinds to the segments or beats. This may entail labelling each beat or section according to the annotation that occurs the closest to its centre.

8.4. Split into training and testing data

- The next step is to divide the input data into training and testing data after loading and preparing it. This is crucial to assess the model's performance on hypothetical data and avoid overfitting from occurring.
- You can split the data into training and testing sets using the "train_test_split" function from the sklearn.
- A function randomly splits the data into training and testing sets, with 80% of the data used for training and 20% used for testing.

8.5. Feature Extraction

The process of turning raw input data into a collection of features that can be fed into a machine learning algorithm is known as feature extraction. Feature extraction is the process of converting the pre-processed ECG signal segments into a set of features that can be utilised as inputs to the deep learning model in the context of arrhythmia detection using deep learning.

Time-domain characteristics: These characteristics of the ECG signal, which include mean, variance, standard deviation, skewness, kurtosis, and waveform morphology, are statistical and temporal. Techniques including peak detection, QRS complex detection, and template matching are used to extract time-domain information. By, examining the temporal and spectral characteristics of the ECG signal, arrhythmia can be picked up from ECG waveforms. An ECG shows the electrical activity of the heart throughout time in a visual form. An electrical event in the heart is represented by a series of waves and intervals that make up the ECG waveform.

The most popular method for detecting arrhythmias from ECG waveforms is to examine the QRS complex's shape, which depicts the depolarization of the heart's ventricles. A ventricular arrhythmia, such as ventricular tachycardia or ventricular fibrillation, can be detected by abnormalities in the QRS complex's duration, amplitude, and morphology, ventricular hypertrophy.

By examining many aspects of the ECG signal, such as the morphology of the QRS complex, the HRV, and other spectral and temporal properties, machine learning techniques, including deep learning, can be used to automatically detect arrhythmia from ECG waveform. These methods can help increase the speed and accuracy of arrhythmia diagnosis as well as uncover arrhythmia patterns that could be challenging to find through visual examination of the ECG waveform.

8.6. Data Augmentation

By producing more synthetic instances based on the original data, a technique known as "data augmentation" can be utilised to expand the amount of training data. The goal is to perform a number of transformations on the initial data to produce new data points that are comparable to the original data points but not exactly the same. This can be especially helpful when there is a shortage of training data because it enables the model to see more instances of the same thing.

8.7. Define deep learning model :cnn

The choice of model depends on several factors such as the size of the dataset, the complexity of the arrhythmia detection task, and the available computational resources. In general, larger and more complex models tend to perform better, but also require more training data and longer training times.

During training, the weights of the filters in the convolutional layers are learned through backpropagation. The objective is to minimize a loss function that measures the difference between the predicted output and the true output. By minimizing this loss, the CNN learns to extract relevant features from the input images and produce accurate predictions on new images.

8.8. LSTM&VGG-16 models

A CNN+LSTM model combines the spatial feature extraction power of a Convolutional Neural Network (CNN) with the sequential modeling capabilities of a Long Short-Term Memory (LSTM) network. The input data is passed through convolutional and pooling layers to extract spatial features. The output is then passed through LSTM layers to model the sequential nature of the data. The final output is generated using fully connected layers, and the model is trained using backpropagation to minimize a loss function.

VGG16 is a deep convolutional neural network with 16 layers that uses 3x3 filters and max-pooling layers to extract features from an input image. It then passes the features through fully connected layers to generate a probability distribution over the possible class labels. VGG16 is trained through backpropagation to minimize a loss function and is effective in a variety of image classification tasks.

8.9. Train

Once the model is compiled, the next step is to train the model on the training data. Training a deep learning model involves iteratively updating the model's parameters (weights and biases) using the training data in order to minimize the loss function.

During training, the model will update its parameters using the training data and evaluate its performance on the validation data after each epoch. The training progress and performance metrics can be monitored using the history object.

The history object contains the loss and metric values for the training and validation data at each epoch. This information can be used to plot learning curves and evaluate the performance of the model.

8.10. Evaluate

Evaluating a deep learning model involves computing its performance metrics on a new set of data that it has not seen before.

9. Result

A comparative study was conducted among the three models and the results obtained are presented below

Table 1 Comparative accuracy study among the three models

Model	Accuracy obtained (%)
CNN	79.3
VGG-16	72.2
CNN+LSTM	96.3

Initially, the model was trained using a CNN architecture, which resulted in an accuracy of 70.2%. To improve the model's performance, more convolutional layers were added, resulting in an accuracy of 79.3%. The CNN architecture was then combined with an LSTM, resulting in the best performance of 96.3%. In contrast, the VGG-16 architecture alone only achieved an accuracy of 72.2%.

10. Conclusion

The detection of arrhythmia using deep learning techniques, such as CNN and CNN-LSTM, has gained significant attention in recent years. The CNN architecture has been widely used in image classification tasks, and it has shown promising results in detecting arrhythmia. The CNN-LSTM architecture combines the strengths of both CNN and LSTM, which can capture both the spatial and temporal features of electrocardiogram (ECG) signals, making it an effective approach for detecting arrhythmia.

Several studies have compared the performance of different deep learning architectures for detecting arrhythmia. One study found that the CNN architecture achieved an accuracy of 70.2% initially. However, by increasing the number of convolutional layers, the accuracy improved to 79.3%. The addition of an LSTM to the CNN architecture resulted in a significant improvement in performance, achieving an accuracy of 96.3%. These results suggest that the CNN-LSTM architecture is highly effective in detecting arrhythmia.

In contrast, the VGG-16 architecture did not perform well in detecting arrhythmia, yielding an accuracy of only 72.2%. The VGG-16 architecture is commonly used for image classification tasks, but it may not be suitable for detecting arrhythmia due to the unique features of ECG signals. Overall, these findings suggest that the CNN-LSTM architecture is a highly effective approach for detecting arrhythmia, outperforming both the CNN and VGG-16 architectures. The use of deep learning techniques in detecting arrhythmia has the potential to improve the accuracy of diagnosis and treatment, leading to better patient outcomes.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest.

References

[1] K. Vayadande et al., "Heart Disease Prediction using Machine Learning and Deep Learning Algorithms," 2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), Greater Noida, India, 2022, pp. 393-401, doi: 10.1109/CISES54857.2022.9844406.

[2] D. P. Yadav, P. Saini and P. Mittal, "Feature Optimization Based Heart Disease Prediction using Machine Learning," 2021 5th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2021, pp. 1-5, doi: 10.1109/ISCON52037.2021.9702410.

[3] R. Deepika, P. B. Srikaanth and R. Pitchai, "Early Detection of Heart Disease Using Deep Learning Model," 2022 8th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2022, pp. 1-4, doi: 10.1109/ICSSS54381.2022.9782295.

[4] M. M. Farag, "A Self-Contained STFT CNN for ECG Classification and Arrhythmia Detection at the Edge," in IEEE Access, vol. 10, pp. 94469-94486, 2022, doi: 10.1109/ACCESS.2022.3204703.

[5] Y. K. Kim, M. Lee, H. S. Song and S. -W. Lee, "Automatic Cardiac Arrhythmia Classification Using Residual Network Combined With Long Short-Term Memory," in IEEE Transactions on Instrumentation and Measurement, vol. 71, pp. 1-17, 2022, Art no. 4005817, doi: 10.1109/TIM.2022.3181276.

[6] Liu, X., Wang, H., Li, Z. and Qin, L., 2021. Deep learning in ECG diagnosis: A review. Knowledge-Based Systems, 227, p.107187.

[7] Keys, A., 1975. Coronary heart disease—the global picture. Atherosclerosis, 22(2), pp.149-192.

[8] Dupre, A., Vincent, S. and Iaizzo, P.A., 2005. Basic ECG theory, recordings, and interpretation. Handbook of cardiac anatomy, physiology, and devices, pp.191-201.

[9] https://en.m.wikipedia.org/wiki/File:SinusRhythmLabels.svg.

[10] Huiskamp, G. and Van Oosterom, A., 1988. The depolarization sequence of the human heart surface computed from measured body surface potentials. IEEE Transactions on biomedical engineering, 35(12), pp.1047-1058.

[11] Sedmera, D. and Gourdie, R.G., 2014. Why do we have Purkinje fibers deep in our heart?. Physiological research, 63.

[12] Fenton, F.H., Cherry, E.M. and Glass, L., 2008. Cardiac arrhythmia. Scholarpedia, 3(7), p.1665.

[13] https://my.clevelandclinic.org/-/scassets/images/org/health/articles/16749-arrhythmia-illustration

[14] Yagel, S., Weissman, A., Rotstein, Z., Manor, M., Hegesh, J., Anteby, E., Lipitz, S. and Achiron, R., 1997. Congenital heart defects: natural course and in utero development. Circulation, 96(2), pp.550-555.

[15] Goodacre, S. and Irons, R., 2002. Atrial arrhythmias. BMJ, 324(7337), pp.594-597.

[16] John, R.M., Tedrow, U.B., Koplan, B.A., Albert, C.M., Epstein, L.M., Sweeney, M.O., Miller, A.L., Michaud, G.F. and Stevenson, W.G., 2012. Ventricular arrhythmias and sudden cardiac death. The Lancet, 380(9852), pp.1520-1529.

[17] Sidhu, S. and Marine, J.E., 2020. Evaluating and managing bradycardia. Trends in cardiovascular medicine, 30(5), pp.265-272.

[18] Ganz, L.I. and Friedman, P.L., 1995. Supraventricular tachycardia. New England Journal of Medicine, 332(3), pp.162-173.

[19] Glazer, A.M., 2022. Genetics of congenital arrhythmia syndromes: the challenge of variant interpretation. Current opinion in genetics & development, 77, p.102004.

[20] [20]Wu, J., 2017. Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. Nanjing University. China, 5(23), p.495.

[21] Lavin, Andrew. "maxdnn: An efficient convolution kernel for deep learning with maxwell gpus." arXiv preprint arXiv:1501.06633 (2015).

[22] https://media.geeksforgeeks.org/wp-content/uploads/20210805213534/max.png

[23] Albawi, S., Mohammed, T.A. and Al-Zawi, S., 2017, August. Understanding of a convolutional neural network. In 2017 international conference on engineering and technology (ICET) (pp. 1-6). Ieee.

[24] Bottou, L., 2012. Stochastic gradient descent tricks. Neural Networks: Tricks of the Trade: Second Edition, pp.421-436.

[25] Graves, A. and Graves, A., 2012. Long short-term memory. Supervised sequence labelling with recurrent neural networks, pp.37-45.

[26] Sherstinsky, A., 2020. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. Physica D: Nonlinear Phenomena, 404, p.132306.

[27] https://miro.medium.com/v2/resize:fit:720/format:webp/1*Mb_L_slY9rjMr8-IADHvwg.png

[28] Gers, F.A., Schmidhuber, J. and Cummins, F., 2000. Learning to forget: Continual prediction with LSTM. Neural computation, 12(10), pp.2451-2471.

[29] Bartolucci, F., Farcomeni, A. and Pennoni, F., 2012. Latent Markov models for longitudinal data. CRC Press.

[30] Qassim, H., Verma, A. and Feinzimer, D., 2018, January. Compressed residual-VGG16 CNN model for big data places image recognition. In 2018 IEEE 8th annual computing and communication workshop and conference (CCWC) (pp. 169-175). IEEE.

[31] Agarap, A.F., 2018. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.

[32] https://miro.medium.com/v2/resize:fit:720/format:webp/1*VPm-hHOM14OisbFUU4cL6Q.png

[33] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K. and Fei-Fei, L., 2009, June. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

[34] Moody, G.B. and Mark, R.G., 2001. The impact of the MIT-BIH arrhythmia database. IEEE engineering in medicine and biology magazine, 20(3), pp.45-50.

[35] Stančin, I. and Jović, A., 2019, May. An overview and comparison of free Python libraries for data mining and big data analysis. In 2019 42nd International convention on information and communication technology, electronics and microelectronics (MIPRO) (pp. 977-982). IEE

[36] Abadi, M., 2016, September. TensorFlow: learning functions at scale. In Proceedings of the 21st ACM

[37] VP, H.V., Rahiman, A. and Abraham, L., 2022. A review on the impact of deep learning in the identification of atrial septal defect and a comparative study on the algorithms employed in the imaging modalities. Journal of Pharmaceutical Negative Results, pp.1922-1930