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

Enhanced Deep Learning Based Non-Invasive Anomaly Detection of ECG Signals with Emphasis on Diabetes

A. K. Ashfauk Ahamed[1], K. Lalitha[2], S. Saravanan[3] and S. MuthuKumar[4]

Submitted: 10/02/2023 **Revised:** 12/04/2023 **Accepted:** 04/05/2023

Abstract: Biomedical signals contain useful information about the activity of different parts of the body. Biomedical signals are basically non in nature. Hence it is very difficult to signals, directly in the time domain, just by observing them. Hence, signal processing techniques are employed to extract important features from these signals for the diagnosis of different diseases.ECG (Electrocardiogram) signal indicates the working of autonomic nervous system (ANS) which regulates the normal rhythm of heart. Like any other bio signal, ECG signal are also non-linear and non-stationary in nature. The analysis of ECG gives a clue to different diseases.The main objective of this paper is to find out methods to diagnose diabetes using heart rate variability (HRV) signals employing deep learning-based Convolution Neural Network (CNN) and also, we used hyper parameter tunning the classifier by Ant colony optimization (ABC) algorithms. And also we discuss and used feature extraction process handled by using Deep Belief Network, so as to get very high accuracy of detection. We devise HOS based machine learning method, then deep learning-based method and then an improvement of our deep learning work to achieve increased accuracy values to achieve automated diabetes detection using HRV signals.

Keywords: *Deep Learning, Heart Rate Variability, Convolution Neural Network, Autonomic Nervous System and Non-Invasive Anomaly Detection.*

1. Introduction

Biosignals are biological signals collected from the human body or, more broadly, from humans. Biosignals are commonly referred to as electrical in nature, although there are nonelectrical biosignals as well. Biosignals have been processed and evaluated primarily by extracting characteristics and then classifying them since their inception. These procedures are carried out by creating computer-aided design (CAD) systems [1]. To detect anomalies, machine learning requires extensive domain knowledge of biosignals and the functioning of the human body. The first step in this direction is proper feature identification, which is followed by categorization [2]. Deep learning is a customized version of machine learning. Deep learning, unlike machine learning, does not require explicit feature engineering. All of these tasks are performed implicitly by the deep learning network's hidden layers, without the involvement of an external researcher. Deep learning architectures are often composed of a large number of hidden layers in a form akin to a 2D matrix. Deep learning can be used to effectively analyze complicated, high-dimensional real-world data. Raw data can be sent directly into these networks [3].

For machine learning to find oddities, it needs to know a lot about biosignals and how the human body works. The first step in this way is to find the right features, and then to put them into groups. Deep learning is a type of machine learning that is tailored to each person. Deep learning, on the other hand, doesn't need clear feature building like machine learning does [4]. All of these jobs are done automatically by the hidden layers of a deep learning network, without the help of an outside expert. Deep learning designs often have a lot of hidden layers and millions of neurons linked to each other in a way that looks like a 2D grid. Deep learning can be used to study complex, high-dimensional data from the real world in a useful way. You can send raw data or data with little signal processing right into these networks [5]. Each layer of the network creates representations that are designed automatically by the deep learning network using a general learning method. This is in contrast to typical machine learning-based neural networks, which are much smaller and less structured than deep learning

[1] Assistant Professor, Dept of Computer Applications, B.S.AbdurRahman Crescent Institute of Science and Technology, Chennai. Email:ashfauk@gmail.com
[2] Assciate Professor, Department of Information Technology, Panimalar Engineering College, Chennai. Email:lalithapecit@gmail.com
[3]Research Scholar, Department of Computer Science and Engineering, B.S.AbdurRahman Crescent Institute of Science & Technology, Chennai. Email:saravanans_cse_jan21@crescent.education
[4] Assistant Professor, Department of Computer Science and Engineering, , St. Joseph's Institute of Technology, Chennai. Email:muthukumars@stjosephstechnology.ac.in

networks, where feature extraction is done by hand. Deep learning networks are often used to process images in two dimensions, but they can also work well with data that only has one dimension. Health care is another place where deep learning can be used [6-7].

A slew of papers on anomaly identification in healthcare have just been released. DL approaches are crucial in a range of activities that use public and private data, such as detecting diabetes anomalies, cardiac arrhythmia, atrial fibrillation, and so on. The ECG (Electrocardiogram) is the sum of millions of cardiac cell potentials. The electrocardiogram (ECG) depicts the operation of the autonomic nervous system (ANS), which regulates the regular rhythm of the heart. ECG begins at the sinoatrial (SA) node and travels through the atria and ventricles. ECG signals, like any other biosignal, are non-linear and non-stationary in nature.

The form of the ECG signal shows how good the cardiac health is. ECG analysis can reveal irregularities directly related to the heart, such as cardiac arrhythmia. It can also provide important information regarding disorders such as diabetes, which may appear to be unrelated to the heart. Diabetes can cause difficulties in many sections of the body, increasing the chance of premature death [8]. Some of the possible problems include kidney failure, heart attack, stroke, leg amputation, visual loss due to diabetic retinopathy, and nerve damage. As diabetes is incurable, the development of efficient diabetic care approaches is critical. Diabetes management relies heavily on early detection [9]. Deep learning techniques are first utilized in this case to identify diabetes using HRV. Bio signal analysis is a critical area of study. Artificial intelligence-based technologies, particularly machine learning and deep learning-based research, are becoming increasingly popular in bio signal analysis. ECG is a vital bio signal that indicates the health of the cardiovascular system. The presence of various anomalies associated with the heart and ANS can be determined non-invasively by analyzing ECG.

This builds on our previous HOS machine learning work. Convolutional neural network (CNN) deep learning architectures were mostly used for the automated identification of diabetes with the best accuracy value. Deep learning networks do not require feature extraction or feature selection in the same way that traditional machine learning-based methods do. In turn, they are integrated in the deep learning network[10].

3.1. Contribution of this study

The main contribution of this paper is described as Deep Learning based Non-Invasive Anomaly Detection of ECG Signals with Emphasis on Diabetes by using Convolution Neural Network (CNN), which is used as classifier and used hyper parameter tunning the classifier by Ant colony optimization (ABC) algorithms.

3.2. Structure of paper

The remaining section of this paper satiated as literature survey in section 2. In next section we have executed the proposed system in section 3. And also the feature extraction in next section. Another section 4, we evaluated the results and discussion. And finally the conclusion is described in section 5.

2. Literature Survey

The objective of this section is to focus on the detailed review of anomaly detection using ECG signals. Concentration is given to the anomaly of diabetes and to HRV derived from ECG. The review provided in the chapter provide the basis for the motivation for the crucial part of our thesis work, which is development of deep learning methodologies for diabetes detection. Below is a detailed study on ECG signal. In recent times, deep learning is slowly replacing machine learning techniques in bio signal analysis. Bahadure et al., [11] implemented the concept of Naturally Inspired Berkeley Wavelet Transformation (BWT) and Support Vector Features Type Based on shape Ex: Area, Shape, Perimeter Based on texture Ex: Sum, Contrast difference, Entropy Based on Intensity Ex: Mean, Median, Standard deviation Machine (SVM). The features like Gray Matter (GM), White Matter (WM), and Cerebrospinal Fluid (CSF) are extracted carefully using SVM. The wavelet and SVM features are implemented to improve the automatic detection accuracy

Ahmed, M. Z., & Mahesh, C. [12] reviewed various methodologies for characterization that are dependent upon the Feed Forward Neural Network (FFNN), Multilayer Perceptron and Back-propagation. Surface highlights were separated to recognize cerebrum tumour grades. They considered 50 MRI samples for feature extraction. The highlights of cerebrum
tumour grades are separated utilizing Grey-Level Co-Occurrence Matrix (GLCM) and Grey Level Run-Length Matrix (GLRM). Separate ideal highlights are separated from the highlights using the fluffy entropy calculation. The classifiers are trained and evaluated based on this evidence from the multiple analyses of brain tumour highlights.

Kabir and Shahnaz [13] discussed setting forth a strategy for ECG picture order by extricating their component utilizing wavelet change and neural organizations. Highlights are extricated from wavelet deterioration of the ECG pictures power. The got ECG highlights are then additionally handled utilizing fake neural organizations. The elements are: mean, middle, greatest, least, range, standard deviation, difference, and mean outright deviation. The presented ANN was prepared by the primary highlights of the 63 ECG pictures of various infections. The experimental outcomes showed that the order precision of the acquainted classifier was up to 92%. The extricated elements of the ECG signal utilizing wavelet disintegration was viably used by ANN in delivering the arrangement precision of 92%.

Shaker et al., [14] introduced another information increase model using the GAN strategy for re-establishing the equilibrium of the dataset. The 2 DL a start to finish strategy and a 2-stage progressive methodology relying on DCNN is used for killing hand designing highlights with the mix of component decrease, grouping, and element extraction to a solitary learning procedure. Nurmaini et al., (2020), DL is introduced in the calibrating and pre-preparing stages for delivering a programmed include portrayal to a multi-class grouping of arrhythmia condition. In the pre-preparing stage, stacked DAE and AE are used to highlight learning; in calibrating stage, DNN is planned as classifiers.

Huang et al., [15]introduced an exact order model dependent on smart ECG using FCResNet. In this presented framework, the MOWPT gives a far-reaching time scaleclearing design and has time invariance includes that are utilized for decaying the genuine ECG sign to sub sign examples of different scales. Then, at that point, the example of five arrhythmia structures is utilized as a contribution to the FCResNet; consequently, ECG arrhythmia types are ordered and perceived.

In Li et al., [16] the morphology and musicality of pulses are converged to 2D information vector to process therefore utilizing CNN which incorporates one-sided dropout and versatile learning rate draws near. The outcome shows the projected CNN module is productive to recognize strange pulses/arrhythmias via mechanized element extraction.

Table 1: A summary of machine learning approaches used for detection HRV parameters

Methodology	Experimental Activity for Extracted Features	Reference
Nonlinear (RQA, CD)	Accuracy is 86%	[17]
HOS	Accuracy is 90.5%	[18]
HOS	Accuracy is 79.93%	[19]
Nonlinear features	Accuracy is 90.0%	[20]
DWT	Accuracy is 92.02%	[21]
EMD related features	Accuracy is 95.63%	[22]

3. Proposed Methodology

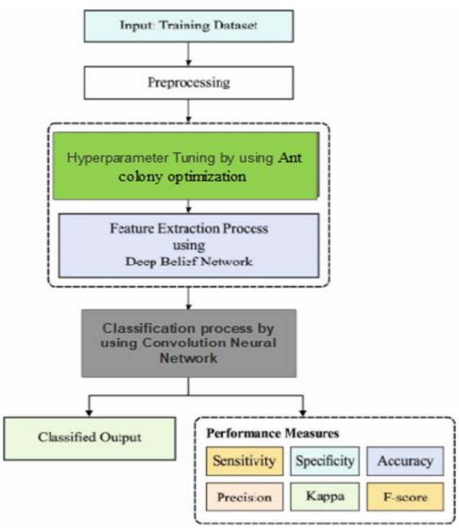

Fig. 1: The Proposed Architecture

3.3. Data and methods

The MIT-BIH arrhythmia database from Physionet is used. The database contains 48 half-hour two-channel ECG recordings from 47 people. Most algorithms, in general, employ the MIT-BIH database in its original format, which is the same data recorded in 48 sequences (101.dat to 124.dat, 200.dat to 234.dat (total 48 sequences with some intermediate numbers missing)). When used to deep learning networks in its original structure, this database has a serious flaw. This is because these 48 sequences mostly comprise of two sorts of data (either normal or aberrant heartbeats). As a result, beat-to-beat dependency in data sequences is quite likely. Another issue is the disparity in baseline voltages between sequences. To address both of these difficulties, we isolated individual heartbeats from database continuous sequences.

The majority of the instances yield 361 samples because the window size is set to 1 second. Because the last sample was duplicated in order to make the total sample values 361 in all examples, some examples contained 360 samples. Examples with annotations other than normal or some form of arrhythmia were eliminated. This is because the goal of this work is to categorize the presented ECG data as normal or abnormal (arrhythmia). Following the excision, we are left with 93521 cases, of which 18482 (19.7%) showed arrhythmia.

3.4. Pre-processing:

ECG signals are typically preprocessed before feeding them into a CNN. This preprocessing may include filtering to remove noise, resampling to ensure a consistent sampling rate, and normalization to bring the signals within a certain range. Network Architecture: The CNN architecture for ECG anomaly detection can be designed based on the specific requirements of the task.

Typically, the network would consist of convolutional layers to capture local patterns and features from the ECG signals. Pooling layers can be used to downsample the feature maps. Additional layers, such as fully connected layers and output layers, can be incorporated for classification or anomaly detection. he CNN is trained using the prepared dataset. During training, the network learns to distinguish between normal ECG signals and those with anomalies associated with diabetes. The training process involves feeding the ECG signals into the network, computing the loss (e.g., binary cross-entropy) between predicted and true labels, and updating the network's weights using backpropagation and an optimization algorithm (e.g., stochastic gradient descent).

3.5. Feature Extraction

The embedding method token is in the biological event trigger text in the data set. This token is turned into a semantic feature of the word. This makes it possible to show how the extraction event causes different anchoring sizes. In this study, word embedding helps find appropriate semantics from the information in the input word. During the training process, the study uses random information, turns on embedded vectors at the top of each entity, and uses annotation entities to insert their purpose.

3.6. Radial Belief Neural Network for Feature Extraction

The term "Radial Belief Neural Network" is not a commonly used or widely recognized term in the field of neural networks and feature extraction. However, I can provide you with information about radial basis functions (RBFs) and how they can be used for feature extraction.

A radial basis function is a mathematical function that takes a vector as input and outputs a scalar value. RBFs are often used in neural networks for tasks such as function approximation, clustering, and classification. The basic idea is that each RBF unit represents a prototype or centre point in the input space, and its output is based on the similarity between the input vector and the centre point.

In the context of feature extraction, RBF networks can be used to transform the original input space into a higher-dimensional space where the extracted features are more discriminative. This transformation is achieved by using the RBFs to compute the activations of the hidden units in the network.

Here's a high-level overview of how a Radial Basis Function Network (RBFN) can be used for feature extraction:

- ❖ Select the number and positions of the RBF units: In this step, you need to determine the number and locations of the RBF units (center points) in the input space. These centre points can be chosen using various methods such as clustering algorithms or heuristics.
- ❖ Compute the activations of the RBF units: For a given input vector, compute the activation of each RBF unit based on the similarity between the input vector and the centre point. This similarity is typically measured using a distance metric such as Euclidean distance or Gaussian similarity.
- ❖ Apply a linear transformation: The activations of the RBF units form a new set of features in a higher-dimensional space. To further process these features, you can apply a linear transformation, such as a weight

matrix multiplication, to obtain the final feature representation.

❖ Use the extracted features for further analysis: Once you have obtained the transformed feature representation, you can use it for various tasks such as classification, regression, or clustering.

❖ It's worth noting that the specific details and variations of RBF networks can vary depending on the application and specific problem you are trying to solve. The overall architecture and training procedure may differ based on the desired outcomes.

❖ While RBF networks can be useful for certain types of problems, it's important to explore other feature extraction methods as well and choose the one that best suits your specific task and data.

3.7. Hyper parameter and network selection

For this work too, we used Keras and TensorFlow as backend. The performance of deep learning algorithms relies heavily on choosing optimal values for the hyper parameters like learning rate. Experiments were conducted to find optimum value for these hyper parameters. Back propagation through time (BPTT) (Werbos 1990) approach is employed for the training of the deep learning networks.

3.8. Selection of hyper parameters

We used moderately sized architectures. The number of units chosen for trials for RNN and for memory blocks of GRU and LSTM were 32, 64 and 128. We run two trails of experiment with a CNN and pooling layer. The number of filters was tried for values of 32, 64, 128 and filter length were tried for values 2, 3 and 5. The number of filters 64 and filter length 3attained highest accuracy in five-fold cross-validation. For the remaining experiments, we used these optimum parameter values which resulted in maximum accuracy (five-fold cross-validation). Trail of experiment is 3, epochs 300 and 64 units/memory blocks. The training cost was also one of the deciding criteria for fixing 64 units/memory blocks. In order to find an optimal learning rate, two trails of experiment were run for all deep learning architecture still 500 epochs are reached. Learning rate was fixed at 0.1 after trying values from 0.001 to 0.5.

Learning Rate Scheduling: Modifying the learning rate during training can help improve convergence and achieve better performance. Techniques like learning rate decay, where the learning rate is reduced over time, and learning rate warm-up, where the learning rate starts small and gradually increases, are commonly used.

Batch Normalization: Batch normalization is a technique that normalizes the inputs of each layer in a neural network to have zero mean and unit variance. It helps stabilize and speed up training by reducing the internal covariate shift. It enables higher learning rates, helps regularize the network, and acts as a form of normalization.

Weight Regularization: Weight regularization techniques, such as L1 and L2 regularization, are used to prevent overfitting. Regularization adds a penalty term to the loss function, discouraging large weight values. L1 regularization promotes sparsity by driving some weights to exactly zero, while L2 regularization encourages small weights.

Dropout: Dropout is a regularization technique where randomly selected neurons are ignored during training. It helps prevent overfitting by reducing co-adaptation between neurons. During training, each neuron's output is temporarily dropped out with a probability, and the remaining neurons must compensate for their absence. This leads to a more robust network.

Early Stopping: Early stopping involves monitoring the validation loss during training and stopping the training process when the validation loss starts increasing. This helps prevent overfitting and allows the model to generalize better by finding the optimal point before it starts overfitting the training data.

Data Augmentation: Data augmentation involves generating additional training data by applying various transformations to the existing data, such as rotation, translation, scaling, flipping, and noise addition. It helps increase the size and diversity of the training set, leading to better generalization.

3.9. Ant colony optimization

Ant Colony Optimization (ACO) is a metaheuristic optimization algorithm inspired by the foraging behavior of ants. It is commonly used to solve combinatorial optimization problems, such as the traveling salesman problem (TSP), routing problems, and scheduling problems. ACO mimics the behavior of ants searching for food and finding the shortest path between their nest and food sources. Here's a general overview of how Ant Colony Optimization works:

where $\vec{P}_i(x)$ describes the sites of i^{th} rats, and limits A and C are intended as shadows:

$$A = R - x \times \left(\frac{R}{Iter_{max}}\right), x = 1,2,3, \dots, Iter_{max} \quad (1)$$

$$C = 2 \times rand \quad (2)$$

The parameters here are two random numbers, R [1, 5] and C [0, 2]. In this context, x represents the current iteration of the optimisation process, whereas Iter_max is the extreme allowed. The optimal solution is updated and stored using Eq. (18).

Words that provide more or less weight to a text's overall emotion are given special consideration using attention models. A typical method for doing this is the Weighted Combination of all hidden States, SAW, which looks like this:

$$a_t = \frac{\exp\,(v^T.\tilde{h})}{\sum_t \exp\,(v.\tilde{h})} \quad (3)$$

$$S_{A_W} = \sum_t a_t h_t \quad (4)$$

where $\tilde{h}_t$ and h are defined as shown in Eqs. (3) and (4) and v is a trainable limit.

- ❖ Problem Representation: The problem to be solved is formulated as a graph, where nodes represent cities, vertices represent connections between cities, and weights on the edges represent the cost or distance between cities. Each city is visited exactly once in the case of the TSP.
- ❖ Initialization: A population of artificial ants is created. Each ant is placed in a random city as the starting point.
- ❖ Construction of Solutions: The ants construct solutions iteratively by moving from one city to another according to certain rules. At each step, an ant selects the next city to visit based on a combination of pheromone trails and heuristic information. The pheromone trails represent the pheromone deposits left by previous ants, and the heuristic information guides the ants based on factors like the distance between cities.
- ❖ Pheromone Update: After all ants have constructed their solutions, the pheromone trails are updated. Ants deposit pheromone on the edges they traverse, with the amount of pheromone being proportional to the quality of the solution. The pheromone trails evaporate over time to avoid stagnation and encourage exploration.
- ❖ Solution Evaluation: Once the construction and pheromone update steps are completed, the quality of the solutions constructed by the ants is evaluated using an objective function specific to the problem being solved. In the case of the TSP, the objective is to minimize the total distance traveled.
- ❖ Iteration: Steps 3 to 5 are repeated for a predefined number of iterations or until a termination condition is met. The iteration process allows the ants to refine their paths, focusing on the most promising areas of the search space.
- ❖ Solution Extraction: At the end of the iterations, the best solution found by the ants, which corresponds to the path with the minimum objective function value, is extracted as the final solution.

Ant Colony Optimization leverages the concept of positive feedback (through pheromone trails) and negative feedback (through pheromone evaporation) to guide the search process towards better solutions. The algorithm benefits from the ability of ants to collectively find good solutions by exploiting their local search experiences and exchanging information through pheromone communication.

ACO has been successfully applied to a wide range of optimization problems and has shown promise in finding near-optimal solutions. Various extensions and modifications to the basic ACO algorithm have been proposed to enhance its performance, such as the inclusion of local search mechanisms, diversification strategies, and adaptive parameter settings.

3.10. Classification

Classification is a fundamental task in deep learning, where the goal is to assign input data to predefined classes or categories. Deep learning models are particularly effective in solving classification problems, thanks to their ability to automatically learn complex features from raw data. Here are some key aspects of classification in deep learning: Neural Network Architectures: Deep learning models for classification tasks typically involve neural network architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). CNNs are well-suited for processing grid-like data, such as images, while RNNs are used for sequential data, such as text or time series.

Deep learning has achieved remarkable success in various classification tasks, including image classification, object detection, natural language processing, sentiment analysis, and more. Researchers and practitioners continuously explore new architectures, optimization techniques, and data augmentation strategies to improve the accuracy and robustness of deep learning models for classification.

3.11. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a class of deep learning models specifically designed for processing grid-like data, such as images or sequences. CNNs have revolutionized computer vision tasks and achieved remarkable success in various domains. Here are some key aspects of Convolutional Neural Networks:

❖ Convolutional Layers: CNNs are composed of multiple layers, typically starting with one or more convolutional layers. Convolutional layers apply a set of filters (also known as kernels) to the input data, extracting local patterns and features. The filters slide (convolve) over the input using a mathematical operation called convolution, producing feature maps that highlight different patterns within the data.

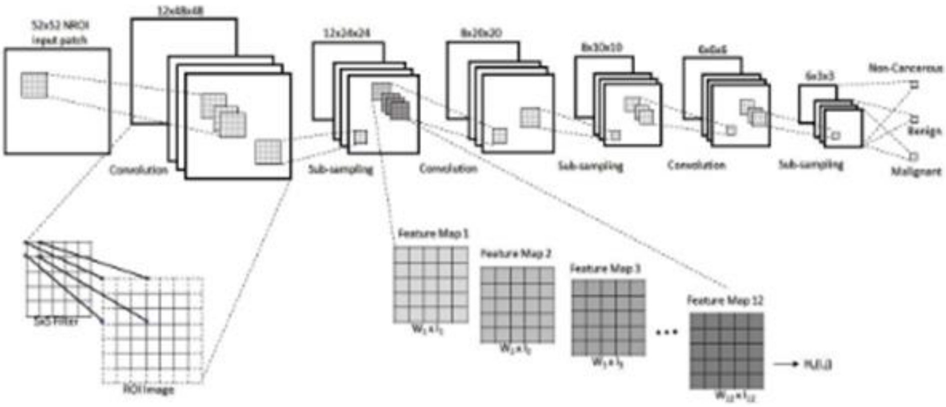

Fig. 2: Proposed CNN Architecture

❖ Pooling Layers: Pooling layers are often added after convolutional layers to downsample the feature maps and reduce the spatial dimensions. Max pooling and average pooling are common pooling operations, where the maximum or average value within a pooling window is retained, respectively. Pooling helps reduce the computational complexity, control overfitting, and achieve some degree of translation invariance.

Fig. 3: Overflow Analysis of Proposed Architecture

❖ Activation Functions: Activation functions introduce non-linearity to the CNN model. The Rectified Linear Unit (ReLU) activation function is widely used in CNNs because of its simplicity and effectiveness. ReLU sets negative values to zero, preserving positive values and facilitating the network's ability to learn complex representations.

❖ Fully Connected Layers: After the convolutional and pooling layers, CNNs often include fully connected layers. These layers connect every neuron in the previous layer to every neuron in the current layer, enabling high-level feature learning and decision-making. Fully connected layers are typically followed by an activation function and a final softmax layer for multi-class classification or a sigmoid layer for binary classification.

❖ Training: CNNs are trained using optimization techniques like stochastic gradient descent (SGD) and backpropagation. The model learns the optimal set of parameters by iteratively updating the weights based on the gradients of the loss function. During training, the network adjusts its parameters to minimize the difference between predicted outputs and the true labels.

Pseudocode of Coevolution Neural Network
$nput: images\ from\ \ dataset, imageSize\ =\ [52\ 52\ 1]$
$AugmentedImageDatastore(imageSize, trainSet, train\ labels, DataAugmentation,$
$imageAugmenter);$
$Augmented\ images\ are\ passed\ as\ input\ to\ convlutional\ layers$
$imageInputLayer(augimds);$
$Layers: convloutional2dLayer(5,12, padding, same)$
$BatchNormalizationLayer$
$reluLayer$
$MaxPooling2dLayer(2, stide, 2)$
$convloutional2dLayer(5,8, padding, same)$
$BatchNormalizationLayer$
$reluLayer$
$MaxPooling2dLayer(2, stide, 2)$
$convloutional2dLayer(5,6, padding, same)$
$BatchNormalizationLayer$
$reluLayer$
$MaxPooling2dLayer(2, stide, 2)$
$FullyConnectedLayer(3)$
$softmaxLayer$
$Training\ options\ :$
$optimizer\ =\ adam$
$Epochs\ =\ 50$
$InitialLearnRate\ =\ 1e-3$
$MiniBatchSize\ = 100$
$Train:\ trainNetwork(auigmids, Layers, options);$
$Classification:\ Ypred\ =\ classify(sample, training)$
$Prediction:\ Yval\ =\ predict(sample, training)$
$Output:\ plotConfusion(Yval, Ypred)$

❖ Transfer Learning: Transfer learning is a popular technique in CNNs, where a pre-trained model is used as a starting point for a new task. The pre-trained model, typically trained on a large dataset, captures generic features and can be fine-tuned or used as a feature extractor for a specific task with limited data. Transfer learning helps in cases where the target dataset is small or when training from scratch is not feasible.

❖ Data Augmentation: Data augmentation is a technique commonly used in CNNs to increase the diversity and size of the training dataset. It involves applying various transformations to the input data, such as random rotation, scaling, flipping, cropping, or adding noise. Data augmentation helps improve the model's ability to generalize by exposing it to a wider range of variations.

Convolutional Neural Networks (CNNs) have been successfully applied to non-invasive anomaly detection in various domains, including electrocardiogram (ECG) signals. One specific application area is the detection of anomalies in ECG signals with an emphasis on diabetes. Here's a brief overview of how CNNs can be utilized for non-invasive anomaly detection of ECG signals related to diabetes:

4. Experimental Setup

Workstation with 3.40 GHz x64 Intel Core i7-6700 CPU, 16 GB of DDR4 RAM, and 4 GB of NVIDIA GetForce graphics card make up the test bed. 256 GB of SSD and 1 TB of HDD make up the storage capacity. Microsoft Windows 10 Professional 64-bit is pre-installed on the solid-state drive. In order to expedite the perfect training and testing process, SSD is utilised to store the datasets and the working settings. This eliminates the mechanical delay imposed by the HDD.

The Anaconda environment has Python version 3.7.15 and all the essential libraries, including NLTK, Stanford NER Tagger, Beautiful Soup, Numpy, Scikit-learn, etc. Our model was trained with a learning rate of 0.001 using the Relu activation function. We have used many measures, to assess performance.

4.1. Performance Measure

Several metrics are used to compare our ideal and similar baseline estimates to the actual outcomes. Below is a comprehensive set of criteria by which ratings may be made:

Accuracy: Accuracy is simply called "accuracy" when referring to test samples.

Precision: Precision denotes to ratio of true positive mockups to the overall amount of false positive samples when discussing predictive value.

Recall: This statistic is useful for measuring the effectiveness of a classifier. An organisational model's sensitivity, often called its true positive rate, if a favourable prediction turns out to be false, recall will ignore it.

F1: This metric is often used in machine learning tasks like classification. It is the mathematical average of how accurate and how many times an estimate was correct.

The existing techniques such as Extreme Learning Machine (ELM), Auto-encoder (AE), RNN (Recurrent Neural Network), and LSTM (long short-term memory networks,) models are considered for the comparison. These techniques are implemented and results are presented.

Table 2: Comparison Analysis of Projected Model with Existing Procedures

Model	Accuracy	Precision	Recall	F1-Score
ELM	80.10	87.21	80.15	80.43
AE	85.71	84.32	85.93	83.45
RNN	92.46	93.32	92.44	91.82
LSTM	94.16	96.17	92.32	92.09
Proposed Methodology	97.62	98.32	96.62	95.53

In the above Table 2 represent that the Comparison analysis of Projected Model with Existing Procedures, in this comparisons analysis, we evaluated the initial model as ELM model reached the accuracy of 80.10 and also, the precision value as 87.21 and the recall value of 80.15 and finally, the F1- score value as 80.43. And another comparisons model of AE model reached the accuracy of 85.71 and also, the precision value as 84.32 and the recall value of 85.93 and finally, the F1- score value as 83.45. And another comparisons model of RNN model reached the accuracy of 92.46 and also, the precision value as 93.32 and the recall value of 92.44 and finally, the F1- score value as 91.82. And another comparisons model of LSTM model reached the accuracy of 94.16 and also, the precision value as 96.17 and the recall value of 92.32 and finally, the F1- score value as 92.09. And another comparisons model of Proposed Methodology 97.62 and also, the precision value as 98.32 and the recall value of 96.62 and finally, the F1- score value as 95.53 respectively. In this comparisons analysis, the proposed model reached the better performance than other compared models.

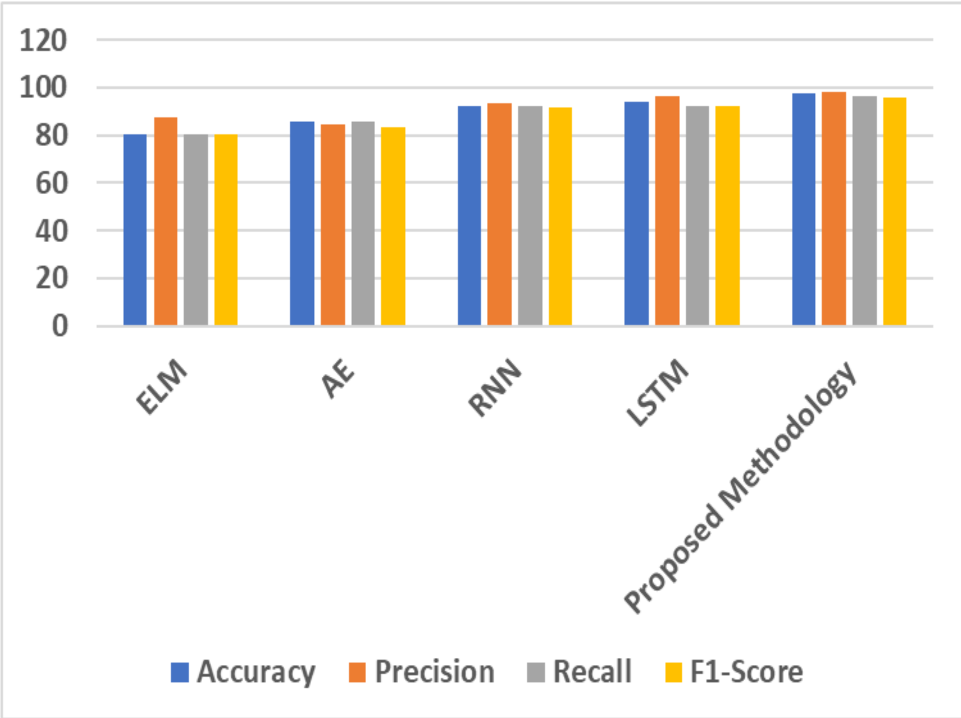

Fig. 4: Graphical Representation of Proposed Model With Other Model Comparisons Performance

In above Figure 4 represent that the graphical representation of proposed model with other model comparisons performance. In this comparisons analysis, the proposed model reached the better performance than other compared models.

5. Conclusion

This method of this paper is to diagnose diabetes using heart rate variability (HRV) signals employing deep learning-based Convolution Neural Network (CNN) and also, we used hyper parameter tuning the classifier by Ant colony optimization (ABC) algorithms. And also we discuss and used feature extraction process handled by using Deep Belief Network, so as to get very high accuracy of detection. We devise HOS based machine learning method, then deep learning-based method and then an improvement of our deep learning work to achieve increased accuracy values to achieve automated diabetes detection using HRV signals.

Reference

[1] Swapna G, Soman KP, Vinayakumar R. Diabetes detection using ecg signals: An overview. Deep Learning Techniques for Biomedical and Health Informatics. 2020:299-327.

[2] Porumb M, Stranges S, Pescapè A, Pecchia L. Precision medicine and artificial intelligence: a pilot study on deep learning for hypoglycemic events detection based on ECG. Scientific reports. 2020 Jan 13;10(1):170.

[3] Woldaregay AZ, Årsand E, Botsis T, Albers D, Mamykina L, Hartvigsen G. Data-driven blood glucose pattern classification and anomalies detection: machine-learning applications in type 1 diabetes. Journal of medical Internet research. 2019 May 1;21(5):e11030.

[4] Shafi I, Aziz A, Din S, Ashraf I. Reduced features set neural network approach based on high-resolution time-frequency images for cardiac abnormality detection. Computers in Biology and Medicine. 2022 Jun 1;145:105425.

[5] K. Muthumayil, R. Karuppathal, T. Jayasankar, B. Aruna Devi, N. B. Prakash, and S. Sudhakar, "A Big Data Analytical Approach for Prediction of Cancer Using Modified K-Nearest Neighbour Algorithm," Journal of Medical Imaging and Health Informatics (2020), Volume 11, Number 8, August 2021, pp. 2184-2189, https://doi.org/10.1166/jmihi.2021.3737

[6] A.Sheryl Oliver, T.Jayasankar, K.R.Sekar, T.Kalavathi Devi, R. Shalini, S. Poojalaxmi and N.G.Viswesh, "Early Detection of Lung Carcinoma Using Machine Learning", Intelligent Automation & Soft Computing, Vol.30, no.3, 2021, pp.755-770. DOI:10.32604/iasc.2021.

[7] J. Jayanthi, M. Kavitha, T. Jayasankar, A. Sagai Francis Britto, N. B. Prakash, Mohamed Yacin Sikkandar and C. Bharathiraja, “Segmentation of Brain Tumor Magnetic Resonance Images Using a Teaching-Learning Optimization Algorithm,”**,** Computers, Materials & Continua, vol. 68, no.3, pp. 4191-4203, 2021

[8] Cordeiro RS. Non-invasive Hyperglycemia Detection Using ECG and Deep Learning (Doctoral dissertation, San Jose State University).

[9] Sannino G, De Falco I, De Pietro G. Non-invasive risk stratification of hypertension: a systematic comparison of machine learning algorithms. Journal of Sensor and Actuator Networks. 2020 Jul 21;9(3):34.

[10] S Stephe, T Jayasankar, “ Biomedical Healthcare Robot Movement Control Using an EEG-Based Brain–Computer Interface with an Optimized Kernel Extreme Learning Machine” was published in a Book In: CRC Press “Robotic Technologies in Biomedical and Healthcare Engineering”, Edited by Deepak Gupta, Moolchand Sharma, Vikas Chaudhary, Ashish Khann (Ed), (2021), pp 157-169. https://doi.org/10.1201/9781003112273

[11] Ahmed, MZ & Mahesh, C 2021, 'A Comprehensive Review on Machine Learning Models for Medical Data Classification', in 2021 2nd International Conference on Smart Electronics and Communication (ICOSEC), pp. 1259-1264

[12] Kabir, MA & Shahnaz, C 2012, 'Denoising of ECG signals based on noise reduction algorithms in EMD and wavelet domains', Biomedical Signal Processing and Control, vol. 7, no. 5, pp. 481-489

[13] Shaker, AM, Tantawi, M, Shedeed, HA & Tolba, MF 2020, 'Generalization of convolutional neural networks for ECG classification using generative adversarial networks', IEEE Access, vol. 8, pp. 35592-35605

[14] Huang, J-S, Chen, B-Q, Zeng, N-Y, Cao, X-C & Li, Y 2020, 'Accurate classification of ECG arrhythmia using MOWPT enhanced fast compression deep learning networks', Journal of Ambient Intelligence and Humanized Computing, pp. 1-18.

[15] Haq, AU, Li, JP, Memon, MH, Nazir, S & Sun, R 2018, 'A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms', Mobile Information Systems, vol. 2018.

[16] Acharya UR, Faust O, VinithaSree S, Ghista DN, Dua S, Joseph P, Thajudin AVI, Janarthanan N, Tamura T, An integrated diabetic index using heart rate variability signal features for diagnosis of diabetes, Computer methods in biomechanics and biomedical engineering, 16: 222-234, 2013.

[17] Wapna G, Ghista DN, Martis RJ, Ang APC and Vinitha Sree S, ECG signal generation and heart rate variability signal extraction: Signal Processing, Featuredetection, and their correlation with cardiac diseases, Journal of Mechanics in Medicine and Biology, 12(4):1240012, 2012.

[18] Ian LW, Lim TC, Automated detection of diabetes by means of higher order spectral features obtained from heart rate signals, Journal of medical imaging and health informatics, 3: 440-447, 2013.

[19] Charya UR, Faust O, Kadri NA, Suri JS, Yu W, Automated identification of normal and diabetes heart rate signals using nonlinear measures, Computers in biology and medicine, 43 (10), 1523–1529, 2013.

[20] Acharya UR, Vidya KS, Ghista DN, Lim WJE, Molinari F, Sankaranarayanan M, Computer-aided diagnosis of diabetic subjects by heart rate variability signals using discrete wavelet transform method, Knowledge-based systems, 81 56–64, 2015.

[21] Pachori RB, Kumar M, Avinash P, Shashank K, Acharya UR, An improved online paradigm for screening of diabetic patients using rr-interval signals, Journal of Mechanics in Medicine and Biology, 16 (01) 1640003, 2016.