Published by
NIGERIAN SOCIETY OF PHYSICAL SCIENCES
Available online @ https://asr.nsps.org.ng/index.php/asr

African Scientific Reports 4 (2025) 359

African Scientific Reports

ECG anomaly detection: a deep learning perspective with LSTM encoders

Adenike Adegoke-Elijah [a,*], Theresa Omolayo Ojewumi [a], Kudirat Oyewumi Jimoh [b]

[a] *Department of Computer Science, Faculty of Computing and Digital Technologies, Redeemer's University, Ede, Osun State, Nigeria*
[b] *Department of Computer Science, Faculty of Computing and Information Technology, Osun State University, Osogbo, Osun State, Nigeria*

Abstract

An electrocardiogram (ECG) is a procedure that measures the heart's electrical activity. Each heartbeat generates an electrical impulse, causing the heart muscle to contract and pump blood. ECG data is a classic example of time series data, where the timing of the upper and lower chambers of the heart is analyzed by medical professionals. The use of unsupervised learning techniques, such as K-Means Clustering and Hierarchical Temporal Memory (HTM) to ECG anomaly detection often results in poor performance due to their dependability on pre-defined features. The performance of a multilayer convolutional neural network adapted to the unsupervised task is also constrained by lack of quality feature extraction and guidance from labelled data. This research proposes a deep learning approach using Long-Short Term Memory (LSTM) Networks for the ECG signal analysis to improve the accuracy of anomaly detection in heartbeats. The proposed method was evaluated on the ECG5000 dataset. The model achieved an impressive accuracy of 99.23%, Specificity of 99.25% and Area under curve (AUC) of 98.00%, thereby outperforming existing models. The results demonstrate that the LSTM Autoencoder-based approach effectively learns expressive representations of ECG sequences, leading to improved performance compared to previous methods.

DOI:10.46481/asr.2025.4.3.359

Keywords: Anomaly, Anomaly detection, LSTM, Autoencoder, Electrocardiogram, Deep learning

Article History :
Received: 14 August 2025
Received in revised form: 23 September 2025
Accepted for publication: 27 October 2025
Published: 29 December 2025

1. Introduction

The most effective method for identifying abnormal cardiac problems is electrocardiography [1]. Electrocardiography (ECG) is a time series data sequence that depicts the process that continuously captures heart electrical signals using electrodes applied to a patient's body. The electrodes are able to recognize skin changes brought on by the heart muscles. An electrocardiogram (ECG) can be used to evaluate the heart's frequency and rhythm, size and locations of its chambers, presence of any abnormalities in the heart muscle cells or conduction system, effects of cardiac medicines, and pacemaker implant functionality [2]. Cardiovascular diseases are among the main causes of premature deaths worldwide [3]. Electrocardiography (ECG), which examines the heart's electrical activities, can be used to identify a number of medical diseases early. Any disruption of the regularity, rate, location, or conduction of the cardiac electric impulse results in heart arrhythmias. In such cases, quick interventions could lead to positive outcomes and

*Corresponding author Tel.: +234-803-419-8862.
Email address: adegoke-elijaha@run.edu.ng (Adenike Adegoke-Elijah)

even save lives [4]. Although, to recognize early signs in patients, constant health monitoring over time is necessary. Medical experts must execute the tedious work of scanning such data for indicatory parts. This herculean task can be resolved by utilizing machine learning algorithms to learn the cases of normal heart activities and automatically detect anomalies when an activity appears significantly different from the earlier types. The term "anomaly detection" describes the process of identifying hidden irregularities in massive amounts of data. It is becoming more important in time-series data for various application domains, such as healthcare [5] and sensor networks [6]. A generic definition of an anomaly is difficult to come up with, even though anomaly detection is quite active study topic. The definition of anomaly is heavily influenced by the application domain and the properties of the time series under consideration. It is frequently essential to use a variety of computational techniques to extract the characteristics of both typical and anomalous behavior or events from data. Due to the large amount of data, the periodic patterns that may be difficult for human eyes to see, the potential for different time scales of arrhythmia, and the variability of heart activities or signals from patient to patient, ECG time-series data presents a challenging task. Time-series anomaly detection has been extensively explored [7] across diverse fields using different techniques. These include supervised and semi-supervised classification-based methods, as well as purely unsupervised approaches. In general, anomaly detection has been implemented using traditional machine learning [8], statistical methods, evolutionary neural network optimization [9], and deep learning techniques [10]. Despite their effectiveness when labeled data are available, supervised algorithms are often less suitable for anomaly detection problems [11], as abnormal instances are rare or difficult to label. Consequently, there has been a resurgence of interest in unsupervised anomaly detection with the emergence of advanced deep generative models such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). For example, Ref. [12] proposed a VAE-based framework for anomaly detection in medical imaging, while Ref. [13] applied VAEs for detecting anomalies in solar photovoltaic generation sequences. Similarly, Ref. [14] employed GANs to identify anomalies in time-series data, demonstrating the flexibility of deep generative models for unsupervised anomaly detection.

A statistical-based method for detecting anomalies in time-series data is the use of ARIMA with multivariate data [15]. Online anomaly detection can also be achieved using Bayesian change point detection methods, which naturally segment time-series data [16]. Other notable frameworks include the Extensible Generic Anomaly Detection System (EGADS), an open-source system developed by Yahoo that combines statistical forecasting and anomaly filtering for scalable detection in high-volume data streams [17]. Within ECG analysis, [18] proposed an adaptive window-based discord detection (AWDD) method for identifying aberrant heartbeats, which improved upon the Brute Force Discord Discovery (BFDD) approach by efficiently distinguishing between normal and abnormal beats in 40-second ECG recordings.

Deep learning models have further advanced ECG anomaly detection by automatically learning complex temporal feature representations without manual feature extraction. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), including Long Short-Term Memory (LSTM) networks, have shown strong performance in modeling temporal dependencies. Ref. [19] used a deep 12-layer CNN for ECG classification, while Ref. [20] combined attention-based LSTM and convolutional layers to capture subtle temporal variations. Ref. [21] introduced a temporal convolutional network for wearable devices aimed at cardiac arrhythmia detection. Earlier, Ref. [22] employed stacked LSTM networks for ECG anomaly detection, demonstrating the power of sequential LSTM encoders for modeling heartbeat patterns.

Recent research trends have focused on hybrid and unsupervised deep learning methods for improved generalization and reduced dependence on labeled datasets. Ref. [2] demonstrated a robust LSTM autoencoder for ECG representation learning and anomaly detection under noisy conditions. Ref. [23] proposed a low-rank attention autoencoder for multi-lead ECG correlation modeling, enhancing spatial and temporal feature capture. Hybrid networks such as CAT-Net, which combine convolutional, attention, and transformer modules, have achieved promising results for arrhythmia classification [24]. Adversarial reconstruction-based frameworks [25] and unsupervised reconstruction/localization methods [26] further highlight the growing interest in semi-supervised ECG anomaly detection.

Despite these advances, several challenges remain, including the generalization of models across diverse patient populations and the efficient detection of anomalies in an unsupervised manner. The present study addresses these gaps by developing an Autoencoder-based Long Short-Term Memory (AE-LSTM) framework that learns compact temporal representations of ECG signals while preserving long-term dependencies for accurate anomaly detection.

This study is organized as follows: Section II describes the LSTM-Autoencoder model, Section III discusses the details of the method used in the study, Section IV summarizes the findings, Section V concludes the study with recommendations.

2. Auto-encoder model

Typically, features or dimensions are extracted or reduced using an autoencoder. Using the data from the initial input sequence $X = \{x_1, x_2, \ldots, x_n\}$, where $x_i \in \mathbb{R}^d$, the function f determines the distinctive sequence of the original data. T denotes the typical sequence of the original sequence, which is defined as $T = \{t_1, t_2, \ldots, t_k\}$, where $t_i \in \mathbb{R}^\ell$. The encoder's output is used as the decoder's input. In accordance with the characteristic sequence T, the decoder reconstructs the original data. The reconstructed data is $\Upsilon = \{y_1, y_2, \ldots, y_k\}$, where $y_i \in \mathbb{R}^d$. Decoding is done to check the validity of the features that were extracted. After the autoencoder has finished its training, it is used to extract the original data's characteristics in order to reveal more of the data's internal structure. Figure 1 depicts the fundamental design of an autoencoder.

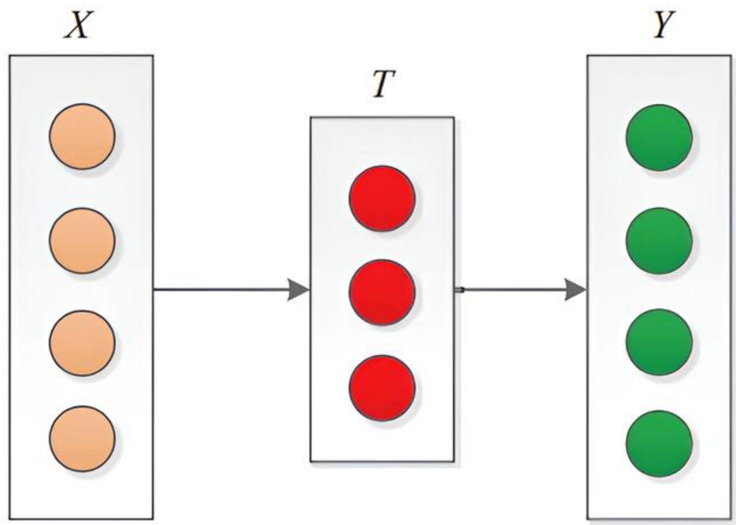

Figure 1: Basic structure of an auto-encoder.

The encoding and decoding processes of the autoencoder are defined as follows:

$$t_i = f(w_t \cdot x_i + b_t), \tag{1}$$
$$y_i = g(w_y \cdot t_i + b_y), \tag{2}$$

where $f(\cdot)$ and $g(\cdot)$ are the sigmoid functions, and w_t, w_y and b_t, b_y are the weights and biases, respectively. As shown in Equation (1), the encoder maps the input x_i to a latent representation t_i. The decoder reconstructs the input as y_i according to Equation (2). The autoencoder is trained by minimizing the reconstruction error:

$$L(X, \Upsilon) = \frac{1}{2}\sum_{i=1}^{n} \|x_i - y_i\|^2. \tag{3}$$

The features extracted from the original data are denoted by T when the disparity between the reconstructed data Υ and the original data X is sufficiently small. The autoencoder extracts these properties, which are then integrated into the detection network to enhance anomaly detection.

The input of the autoencoder is defined as $X_u = \{x_{u1}, x_{u2}, \ldots, x_{um}\}$ and $X_d = \{x_{d1}, x_{d2}, \ldots, x_{dm}\}$, where $x_{ui}, x_{di} \in \mathbb{R}^d$. The typical sequence is represented as $Z_t = \{z_1, z_2, \ldots, z_m\}$, with $z_i \in \mathbb{R}^\ell$.

3. AE-LSTM model

The extracted properties from the autoencoder are incorporated into the LSTM network's input. The input is divided into two categories: the features Z_t and the historical flow data x_t, where the current position is expressed as $X = \{x_1, x_2, \ldots, x_m\}$ with $x_i \in \mathbb{R}^d$.

The LSTM network consists of three primary gates. The forget gate determines which information to discard, the input gate selects what new information to add, and the output gate determines the hidden state. The candidate cell input $\bar{C}_t$ is combined with the previous cell state to update C_t. The activation functions used are:

$$\text{sigmoid}(x) = \frac{1}{1+e^{-x}}, \quad \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}.$$

The LSTM computations are formulated as follows:

$$f_t = \sigma(w_{f1} \cdot x_t + w_{f2} \cdot z_t + w_{f3} \cdot h_{t-1} + b_f), \tag{4}$$
$$i_t = \sigma(w_{i1} \cdot x_t + w_{i2} \cdot z_t + w_{i3} \cdot h_{t-1} + b_i), \tag{5}$$
$$\bar{C}_t = \tanh(w_{c1} \cdot x_t + w_{c2} \cdot z_t + w_{c3} \cdot h_{t-1} + b_c), \tag{6}$$
$$C_t = f_t \cdot C_{t-1} + i_t \cdot \bar{C}_t, \tag{7}$$
$$o_t = \sigma(w_{o1} \cdot x_t + w_{o2} \cdot z_t + w_{o3} \cdot h_{t-1} + b_o), \tag{8}$$
$$h_t = o_t \cdot \tanh(C_t). \tag{9}$$

In this setup, Equation (4) represents the forget gate, Equation (5) the input gate, Equation (6) the candidate cell state, Equation (7) the updated cell state, Equation (8) the output gate, and Equation (9) the final hidden state. These equations collectively define the temporal dynamics and memory mechanism of the AE-LSTM model.

4. Methodology

4.1. Data collection

The ECG5000 dataset is a publicly accessible ECG database introduced by Yanping Chen and Eamonn Keogh in the UCR Time Series Classification Archive. It consists of 5,000 heartbeat sequences, each represented by 140 time steps recorded from an individual patient. These sequences include both normal and abnormal temporal patterns. The dataset comprises five distinct classes of beats labeled from 0 to 4, as summarized below:

- Class 0 — Normal Sinus Rhythm (N): Represents a healthy heartbeat, typically ranging from 50 to 100 beats per minute.
- Class 1 — Premature Ventricular Contractions (PVC): Characterized by "skipped" or early beats that disrupt the heart's normal rhythm, a common form of arrhythmia.
- Class 2 — Supraventricular Premature Beats (SP): Premature beats originating in the atrial myocardium, occurring in both healthy individuals and patients with varying cardiac conditions.
- Class 3 — R-on-T Premature Ventricular Contractions (R-on-T): Occurs when an ectopic beat overlays the T wave of the preceding beat, potentially leading to sustained arrhythmias.
- Class 4 — Unclassified Beats (UB): Represents beats that do not conform to any of the above categories, possibly due to signal noise, rare arrhythmias, or recording artifacts.

In this study, Class 0 signals were treated as normal heartbeats, while Classes 1–4 represented different types of arrhythmias. The dataset was normalized using Min–Max scaling and reshaped to match the LSTM input format of `(batch_size, 140, 1)` for multiclass classification.

Figure 2 depicts the morphological heterogeneity of each class.

The AE-LSTM model was implemented using Python 3.11 and TensorFlow 2.11. Training was performed in the Google Colab environment to leverage GPU acceleration. The ECG5000 dataset was preprocessed using normalization and a data resampling technique. The signals were normalized using Min-Max scaling to ensure consistent input ranges. After normalization, the data were reshaped to fit the input shape required by the LSTM layers as `(batch_size, 140, 1)`.

4.2. Proposed model architecture

The proposed architecture is composed of an encoder, a latent space, and a decoder. The encoding process, which consists of two LSTM layers, receives the ECG signals as a time-series sequence and compresses them into a lower-dimensional representation known as the latent space, with the objective of detecting the important features in the input data.

The first LSTM layer captures local temporal dependencies within the ECG signals and consists of 64 units, which are sufficient to learn meaningful patterns without overfitting. The second LSTM layer captures more abstract and higher-level features of the ECG signals, with the number of units reduced to 32 to create a compact latent representation. A dropout rate of 0.2 is applied to each of the LSTM layers to prevent overfitting, and the `tanh` activation function is used in both layers. The final LSTM layer outputs a fixed-size vector known as the latent space representation, which is a compressed version of the original input sequence. This vector contains the significant features that have been learned by the model to represent the input data.

The decoding process includes a repeat vector layer, which repeats the latent vector 140 times to match the required sequence length. The decoding process also comprises two LSTM layers. The first LSTM layer has 32 units, uses the `tanh` activation function, and applies a dropout of 0.2. This layer begins the reconstruction process from the latent space representation and mirrors the complexity of the second LSTM layer in the encoder. The second LSTM layer has 64 units, uses the `tanh` activation function, and also applies a dropout of 0.2. It continues the reconstruction of the sequence with the goal of recreating the original input signal.

Following this is a TimeDistributed Dense layer, which serves as the output layer of the model. This layer consists of 1 unit, since the ECG signal is a univariate time series, and uses a linear activation function. The output of this layer is the reconstructed sequence, which is expected to closely match the original input sequence if the autoencoder has effectively learned the data patterns. The diagrammatic representation of the proposed model is shown in Figure 3.

The model used the Mean Squared Error (MSE) loss function, which measures the difference between the original and the reconstructed output. The Adam optimizer was employed due to its adaptive learning rate and computational efficiency.

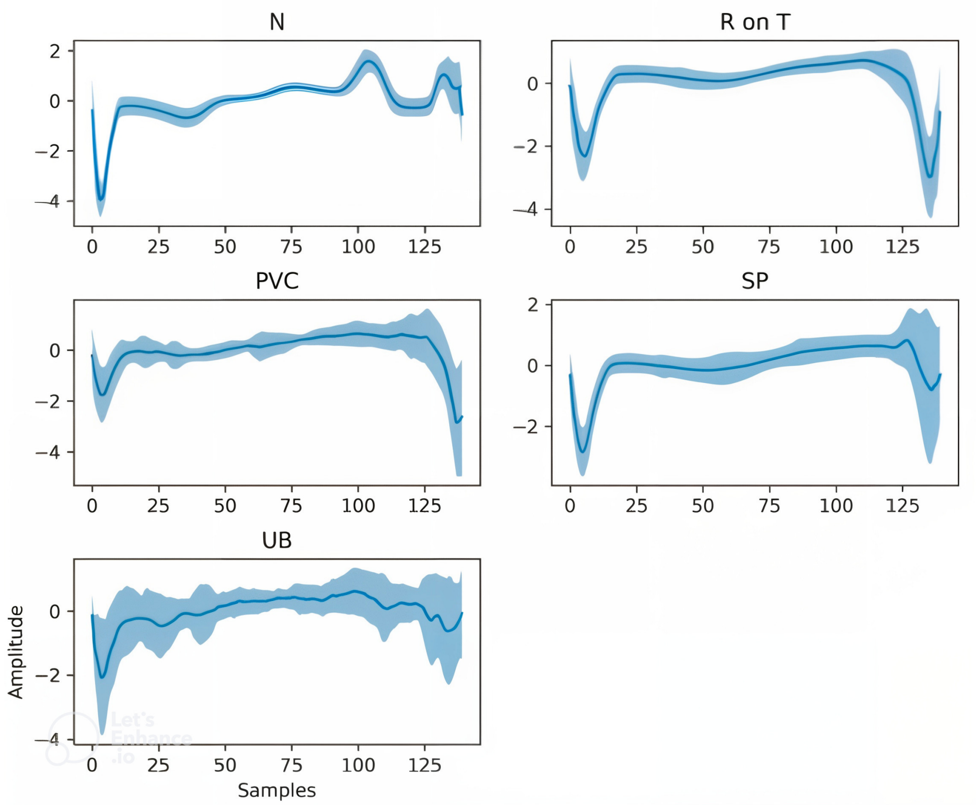

Figure 2: Classes morphological variability on ECG5000 dataset.

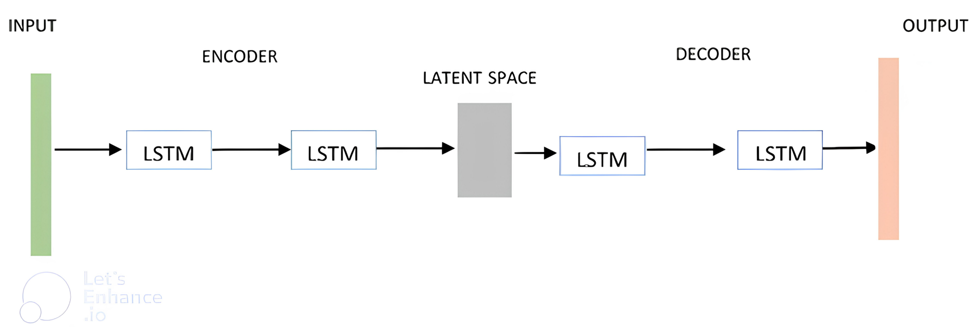

Figure 3: The proposed model.

4.3. *Model training and testing*

The model was trained using the PyTorch framework on the Google Collaboratory cloud platform to facilitate access to Graphical Processing Units (GPUs) for training and testing. The AE-LSTM model was trained on batches of size 32 using the Adam optimizer with a learning rate of 1×10^{-3} for 50 epochs.

Since the task involves anomaly detection, the normal class (class 0) was isolated from the dataset to train the autoencoder. The goal of training is for the autoencoder to effectively reconstruct normal signals with low reconstruction error, while producing higher reconstruction errors for anomalous signals. The normal ECG data was split into a training set (80%) and a validation set (20%). The training set was used to train the model, while the validation set was used to monitor model performance and prevent overfitting.

During training, the model processes each batch of the training data and produces reconstructed outputs. The Mean Squared Error (MSE) loss was computed by comparing the reconstructed output with the original input. Figure 4 displays the AE-LSTM's reconstruction performance, plotted as graphs.

The first three graphs show the AE-LSTM (yellow line) almost perfectly reconstructing the signal (blue line), indicating that the AE-LSTM understands the sequence, and thus, these would be classified as normal heartbeats. The subsequent set of graphs

Figure 4: AE-LSTM sequence reconstruction performance plot.

Table 1: Performance metrics of the AE-LSTM model.

Metric	Accuracy	Specificity	AUC
Value	99.23%	99.25%	98.00%

show the AE-LSTM performing poorly at reconstructing the signal. This is a desirable outcome, as it indicates that the AE-LSTM does not understand the sequence—having been trained only on normal heartbeat sequences—therefore, these would be classified as anomalous heartbeats.

5. Results and discussion

The trained model was evaluated using a test dataset that comprises both anomalous and normal ECG signal to check how the model performs well when exposed to unseen data. The ECG5000 dataset contains a separate test file used to test the model, and contains 500 test samples. The model was tested for accuracy, specificity and Area Under Curve (AUC) using the total number of ECG signals predicted as true positive, true negative, false positive and false negative. Accuracy checks the proportion of correctly classified instances (true positive and true positive) out of the total number of occurrences. Specificity, otherwise known as the true negative rate, it examines the number of actual negatives accurately acknowledged by the model. Area under curve measures the performance of binary classifiers, such as the one in this model and provides an aggregate evaluation of the model performance across all thresholds. Out of the test samples, the evaluation shows 99 true positive, 397 true negative, 3 false positive and 1 false negative. The performance of the auto-encoder model is summarized in Table 1.

The proposed AE-LSTM model was evaluated against existing models that used the same ECG5000 dataset for anomaly detection. Table 2 presents the experimental findings, showing that AE-LSTM outperforms the Variational Autoencoder (VAE) [19], attention-based LSTM-CNN (AttLSTM-CNN) [27], and temporal convolutional network (TCN) [15]. Specifically, the AE-LSTM achieved a performance improvement of 2.18%, 11.49%, and 16.15% over these respective models. Unlike attention-based LSTM-CNNs, which rely heavily on labeled data, AE-LSTM learns compact temporal representations in an unsupervised manner, capturing both local and long-range dependencies in ECG sequences. Statistical and TCN-based methods, while computionally efficient, lack the capacity to capture complex temporal dependencies. These results demonstrate that the AE-LSTM framework provides a robust and efficient approach for ECG anomaly detection, combining the benefits of autoencoding and temporal modeling while reducing dependence on labeled data.

These findings indicate that by leveraging AutoEncoders and LSTM networks, the AE-LSTM model can effectively learn temporal dependencies in time-series data, thereby considerably enhancing anomaly classification performance.

Table 2: Comparison of AE-LSTM with existing models on ECG5000 dataset

Model	Accuracy (%)
AutoEncoder-LSTM	99.23
Variational Autoencoder	97.11
Statistical based Method	89.00
Attention-based LSTM-CNNs	85.43

6. Conclusion

In this paper, an unsupervised method which depends on normal data for training of the model was proposed for ECG anomaly detection. A LSTM and an AutoEncoder were used as the suggested method to extract feature representation for time series classification. It uses an LSTM's forget cells to tap into its capacity to learn order dependence in order to automatically pick up on long-term temporal dependency aspects. In this paper, we describe the results of an unsupervised anomaly detector appropriate for ECG signals. The results are promising, but there is certainly space for improvement. Possible future work can carry out further analysis to consider the local reconstruction score exploring temporal related ones and not only amplitude related distances. This could help in differentiating other types of anomalies from normal heartbeats. A future implementation can exploit the capabilities of the AE-LSTM for detection other domains (e.g., noise, engine and signal corruption) so it can learn normal and abnormal domain-patterns.

Data availability

The dataset used in this study, *ECG5000*, is publicly available from the UCR Time Series Classification Archive at https://timeseriesclassification.com/description.php?Dataset=ECG5000. The dataset was preprocessed prior to model training, as described in the Methodology section. No proprietary or restricted data were used.

References

[1] K. Nezamabadi, N. Sardaripour, B. Haghi & M. Forouzanfar, "Unsupervised ECG analysis: a review", IEEE Reviews in Biomedical Engineering **16** (2023) 208. https://doi.org/10.1109/RBME.2022.3154893.

[2] I. Farady, V. Patel, C. Kuo & C.-Y. Lin, "ECG anomaly detection with LSTM-autoencoder for heartbeat analysis", Proceedings of the IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 2024, pp. 1–5. https://doi.org/10.1109/ICCE59016.2024.10444327.

[3] P. Rani, R. Kumar, A. Jain, R. Lamba, R. K. Sachdeva, K. Kumar & M. Kumar, "An extensive review of machine learning and deep learning techniques on heart disease classification and prediction", Archives of Computational Methods in Engineering **31** (2024) 3331. https://doi.org/10.1007/s11831-024-10075-w.

[4] S. Martin *et al.*, "2024 heart disease and stroke statistics: a report of US and global data from the American Heart Association", Circulation **149** (2024) 347. https://doi.org/10.1161/CIR.0000000000001209.

[5] M. C. Chuah & F. Fu, "ECG anomaly detection via time series analysis", Proceedings of ISPA Workshops, Niagara Falls, Canada, 2007, pp. 123–135. https://link.springer.com/chapter/10.1007/978-3-540-74767-3_14.

[6] A. S. Kumar, S. Raja, N. Pritha, H. Raviraj, R. B. Lincy & J. J. Rubia, "An adaptive transformer model for anomaly detection in wireless sensor networks in real-time", Measurement: Sensors **25** (2023) 100625. https://doi.org/10.1016/j.measen.2022.100625.

[7] K. Shaukat, T. M. Alam, S. Luo, S. Shabbir, I. A. Hameed, J. Li, S. K. Abbas & U. Javed, "A review of time-series anomaly detection techniques: a step to future perspectives", Advances in Information and Communication **1** (2021) 865. https://doi.org/10.1007/978-3-030-73100-7_60.

[8] A. B. Nassif, A. A. Abd El-Latif, M. M. Abd El-Latif, M. A. Hussein & A. M. Abd El-Latif, "Machine learning for anomaly detection: a systematic review", IEEE Access **9** (2021) 78658. https://doi.org/10.1007/s42979-025-04352-z.

[9] S. Sarvari, N. F. M. Sani, Z. M. Hanapi & M. T. Abdullah, "An efficient anomaly intrusion detection method with feature selection and evolutionary neural network", IEEE Access **8** (2020) 70651. https://doi.org/10.1109/access.2020.2986217.

[10] M. S. Elsayed, A. Mohamed & H. Aly, "Network anomaly detection using LSTM based autoencoder", Proceedings of the 16th ACM Symp. QoS and Security for Wireless and Mobile Networks, Alicante, Spain, 2020, pp. 37–45. https://doi.org/10.1145/3416013.3426457.

[11] X. Xia, J. Wang, H. Li, H. Zhang & Y. Li, "GAN-based anomaly detection: a review", Neurocomputing **493** (2022) 497. https://doi.org/10.1016/j.neucom.2021.12.093.

[12] M. Elbattah, M. Boussaid & S. Belkadi, "Variational autoencoder for image-based augmentation of eye-tracking data", Journal of Imaging **7** (2021) 83 2021. https://doi.org/10.3390/jimaging7050083.

[13] M. Ibrahim, A. Khan, M. Javed & S. U. Rehman, "Machine learning schemes for anomaly detection in solar power plants", Energies **15** (2022) 1082. https://doi.org/10.3390/en15031082.

[14] W. Cheng, Z. Zhao, Y. Luo, X. Li & H. Zhang, "Anomaly detection for internet of things time series data using GANs with attention mechanism in smart agriculture", Frontiers in Plant Science **13** (2022) 890563. https://doi.org/10.3389/fpls.2022.890563.

[15] K. Islam & A. Raza, "Forecasting crime using ARIMA model", arXiv preprint (2020) [Online]. https://arxiv.org/abs/2003.08006.

[16] Y. M. Zhang, L. Y. Li, Y. Q. Ni & H. F. Lam, "Anomaly detection of structural health monitoring data using the MLE-based Bayesian dynamic linear model", Structural Health Monitoring **20** (2021) 2936. https://doi.org/10.1177/1475921720977020.

[17] M. Wu, Z. Luo, Y. Zheng & X. Hu, "A study on Arrhythmia via ECG signal classification using CNN", Frontiers in Computational Neuroscience **14** (2021) 564015. https://doi.org/10.3389/fncom.2020.564015.

[18] H. Asif & T. Y. Choe, "Abnormal electrocardiogram signal detection based on the BiLSTM network", International Journal of Contents **18** (2022) 1. https://doi.org/10.5392/IJoC.2022.18.2.068.

[19] P. Matias, J. P. Teixeira & A. Neves, "Robust anomaly detection in time series through variational autoencoders and a local similarity score", Proceedings of 14th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2021) – BIOSIGNALS, 2021, pp. 91–102. https://doi.org/10.5220/0010320500002865.

[20] R. V. Lakshmi & S. Radha, "Time series classification using a†tttention-based LSTM and CNN", Proceedings of International Conference on Data Science, Agents & AI (ICDSAAI), Chennai, India, 2023, pp. 1–7. https://doi.org/10.1109/ICDSAAI59313.2023.10452629.

[21] T. M. Ingolfsson, S. Gudmundsson, H. M. Sigurthorsson & K. Kristjansson, "ECG-TCN: wearable cardiac arrhythmia detection with a temporal convolutional network", Proceedings of IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS), Washington, DC, USA, 2021, pp. 1–4. https://doi.org/10.1109/aicas51828.2021.9458520.

[22] M. Thill, S. Däubener, W. Konen, T. Bäck, P. Barancíková, M. Holena, T. Horvat, M. Pleva & R. Rosa, "Anomaly detection in electrocardiogram readings with stacked LSTM networks", Proceedings of 19th International Conference on Information Technologies – Applications and Theory (ITAT), Donovaly, Slovakia, 2019 pp. 17–25. http://ceur-ws.org/Vol-2473/paper10.pdf.

[23] S. Zhang, Y. Fang & Y. Ren, "ECG autoencoder based on low-rank attention", Scientific Reports **14** (2024) 12823. https://doi.org/10.1038/s41598-024-63378-0.

[24] M. R. Islam, M. Qaraqe, K. Qaraqe & E. Serpedin, "CAT-Net: convolution, attention, and transformer based network for single-lead ECG arrhythmia classification", Biomedical Signal Processing and Control **93** (2024) 106211. https://doi.org/10.1016/j.bspc.2024.106211.

[25] L. Shan, Y. Li, H. Jiang, P. Zhou, J. Niu, R. Liu, Y. Wei, J. Peng, H. Yu, X. Sha & S. Chang, "Abnormal ECG detection based on an adversarial autoencoder", Frontiers in Physiology **13** (2022) 961724. https://doi.org/10.3389/fphys.2022.961724.

[26] A. Jiang, C. Huang, Q. Cao, S. Wu, Z. Zeng, K. Chen, Y. Zhang & Y. Wang, "Multi-scale cross-restoration framework for electrocardiogram anomaly detection", Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention, (MICCAI), Vancouver, BC, Canada, 2023, pp. 87–97. https://doi.org/10.1007/978-3-031-43907-0_9.

[27] Q. Du, W. Gu, L. Zhang & S. L. Huang, "Attention-based LSTM-CNNs for time-series classification", Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems (SenSys), Shenzhen, China, 2018, pp. 410–411. https://doi.org/10.1145/3274783.3275208.