Journal of Artificial Intelligence and Data Science (JAIDA)
e-ISSN: 2791-8335
https://dergipark.org.tr/pub/jaida
Vol. 4, No. 2, pp. 79-86, 2024
Research Article
Received: 09/11/2024 Accepted: 14/12/2024

Time Series Prediction of Heart Rate Using Deep Learning Models

Emir EVCİL [1*]

Abstract
Cardiovascular diseases are among the leading causes of mortality worldwide and represent a significant global health burden, affecting millions of individuals each year. Early diagnosis of these diseases is critical not only for improving patient survival rates but also for ensuring the economic sustainability of healthcare systems. Heart rate values serve as essential biological indicators, providing important insights into cardiovascular health and offering potential utility in early diagnosis. In this study, conducted a comprehensive time series analysis to predict the next 5-minute heart rate values based on a 3-minute segment of pulse data collected from healthy individuals. Employed four deep learning models—Recurrent Neural Network (RNN), Bidirectional Long Short-Term Memory (BI-LSTM), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM)—to analyze the constructed dataset. The predictive performances of these models were rigorously compared using the Root Mean Square Error (RMSE) metric, which serves as a reliable measure of accuracy in regression tasks. Findings indicate that deep learning techniques, particularly LSTM and its variants, hold significant promise for enhancing the accuracy of heart rate predictions. This study underscores the potential of these advanced methodologies in the early diagnosis of cardiovascular diseases, aiming to offer new perspectives for the development of clinical decision support systems that could ultimately improve patient outcomes and optimize healthcare delivery.
Keywords: BiLSTM; GRU; Heart Rate Prediction; LSTM; RNN; Time Series.

1. Introduction

Cardiovascular diseases (CVDs) represent one of the most pressing health challenges globally, contributing significantly to morbidity and mortality rates. The World Health Organization (WHO) estimates that CVDs account for approximately 32% of all global deaths, making early diagnosis and intervention crucial for improving patient outcomes [1]. These conditions often manifest with subtle symptoms, and traditional diagnostic methods may not always provide timely insights. Thus, there is an urgent need for innovative approaches to facilitate early detection.

Heart rate, as a vital physiological parameter, serves as an essential indicator of cardiovascular health. Accurate and continuous monitoring of heart rate values can provide valuable information that aids in the early diagnosis of potential cardiovascular issues [2]. Recent advancements in machine learning, particularly in deep learning techniques, offer promising avenues for analyzing complex time series data, such as heart rate measurements [3].

The heart rate signal is a typical time series [4]. A time series is a series of data points indexed in chronological order. Effective forecasting of time series enables better use of available information for analysis and decision making. Its wide range of applications includes but is not limited to clinical medicine, financial forecasting , traffic flow forecasting, human behaviour forecasting and other fields [5]. Unlike other predictive modelling tasks, time series increases the complexity of sequence dependencies between input variables. Therefore, how to build a suitable predictive model for real-time forecasting tasks by fully exploiting the complex sequence dependencies is an important issue [6].

Importantly, time series data, including heart rate values, often exhibit non-stationarity, meaning their statistical properties, such as the mean and variance, can change over time. This non-stationary behavior can pose significant challenges for modeling and forecasting, as many traditional time series analysis techniques assume stationarity and may not perform well when this assumption is violated [7]. For instance, physiological signals can be influenced by a variety of factors, including emotional states, physical activity levels, and underlying health conditions, all of which can lead to fluctuations in heart rate that reflect complex dynamics rather than stable patterns. Consequently, failing to account for non-stationarity in these signals can result in inaccurate predictions and misinterpretations of the underlying physiological processes. Therefore, developing methodologies that can effectively handle non-stationary time series data is crucial for enhancing the reliability of cardiovascular health monitoring and interventions.

*Corresponding author

Emir EVCİL*; Izmir Katip Celebi University, Faculty of Engineering and Architecture, Electrical Electronic Engineering Department, Türkiye; e-mail: 190403008@ogr.ikcu.edu.tr; 0009-0004-4089-6638

Using machine learning methods, non-linear prediction models can be built based on large amounts of historical data. In fact, through repeated training iterations and learning approaches, machine learning models can obtain more accurate predictions than traditional statistical-based models. Typical methods include tree-based ensemble learning methods such as support vector regression or kernel-based classification and artificial neural multiagent (ANN) with strong nonlinear function approximation, and gradient-enhanced regression or decision tree. However, since the above method lacks efficient handling of sequence dependencies between input variables, its effectiveness in time series forecasting tasks is limited [8].

Recurrent neural networks (RNN) are often regarded as the most efficient method of time series forecasting. In fact, RNN is an artificial neural network in which the nodes are connected in a loop and the internal state of the network can exhibit dynamic timing behaviour. However, as the length of the process time series increases, problems such as gradient disappearance often arise during the training of RNNs using conventional activation functions such as tanh or sigmoid functions, which limit the prediction accuracy of RNNs. The Long and Short Term Memory Unit (LSTM) is based on a simple RNN that solves the memory and forgetting problems by adding some multi-threshold gates. Therefore, LSTM and Gated Loop Unit (GRU) address to some extent the limited ability to deal with long-term dependencies. These methods have been successfully applied to many sequential learning problems such as machine translation [9]. In general, LSTM is considered one of the most advanced methods for dealing with time series forecasting problems. Inspired by cognitive neuroscience, some researchers have incorporated attentional mechanisms into the encoding-decoding framework [10]. Attentional mechanisms can better select input sequences and encode semantics in long-term memory to improve the information processing capabilities of neural multimodality. Recently, attention mechanisms have been widely used and perform well in many different types of deep learning tasks, such as image captioning, visual question answering and speech recognition. Specifically, most research work [11] is usually done by adding an attention layer to the encoding-decoding framework.

Thanks to the effective performance of the LSTM model in time series analysis, it is widely applied to establish long-term relationships in heart rate data and abstract high-dimensional features. For example, Haijun et al. proposed an LSTM-BiLSTM-Att model that combines LSTM, BiLSTM, and an attention mechanism with a fully connected neural network, achieving a notable RMSE of 1.729 in heart rate prediction tasks [12]. Similarly, Haowei et al. emphasized the superiority of transformer-based models for predicting cardiovascular health data, as compared to traditional methods like ARIMA and Prophet, as well as other deep learning approaches [13]. Additionally, Staffini et al. (2022) demonstrated the challenges in predicting heart rate time series due to their nonlinear and non-stationary nature. Their study compared three different prediction models: Autoregressive Model, Long Short-Term Memory Network, and Convolutional Long Short-Term Memory Network, highlighting that the Autoregressive Model consistently outperformed the others across different environments, achieving an average absolute error of 2.069, which was better than the results from the LSTM and ConvLSTM models [14].

By demonstrating the effectiveness of deep learning models in predicting heart rate values, this research contributes to the existing body of knowledge in e-health and underscores the transformative potential of machine learning technologies in enhancing cardiovascular care. Specifically, in this study, we collect data from a pulse oximeter at 1-second intervals, enabling a comprehensive time series analysis. We evaluate the performance of various models, including LSTM, BiLSTM, RNN, and GRU, to assess their effectiveness in capturing the underlying patterns of heart rate variability. The subsequent sections of this paper will detail methodology, present the results of analysis, and discuss the implications of findings for future research and clinical practice.

2. Method

In this study, proposed a deep learning-based system aimed at predicting heart rate five minutes in advance, leveraging the strengths of neural network architectures to handle the temporal dependencies inherent in heart rate data. This predictive model is designed to assist healthcare professionals by providing real-time insights into the patient's cardiovascular trends, which could be instrumental for early intervention in clinical settings. The model development is structured through five essential stages to ensure accurate and reliable predictions:

- Data collection
- Data pre-processing
- Data splitting
- Training and optimization of the model

2.1. Data Collection

Heart rate data was collected continuously over a 3-hour period using a MAX30100 pulse oximeter sensor connected to a Raspberry Pi 3 microcontroller. Measurements were taken at a rate of one sample per second, resulting in a dataset of 10,597 entries.

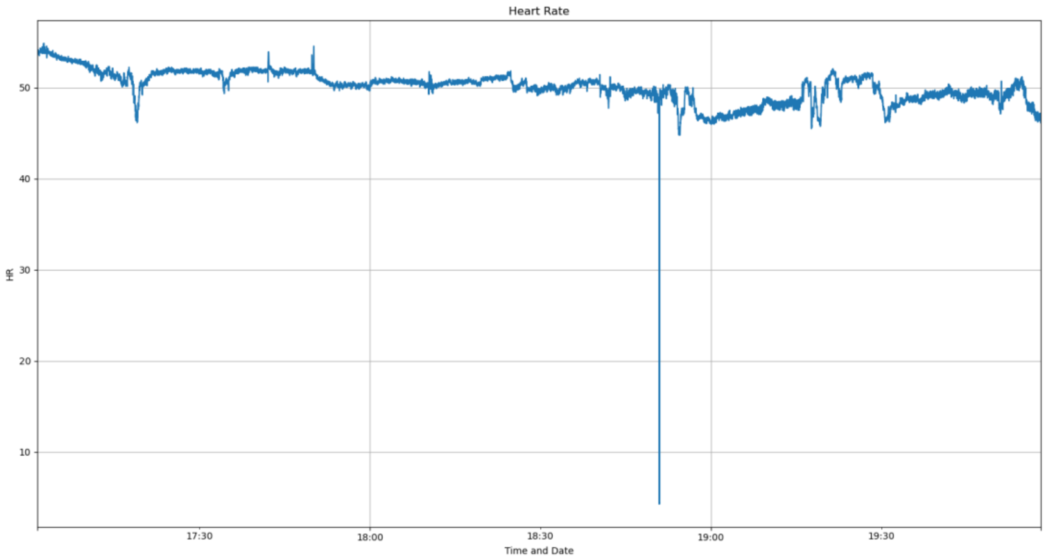

Figure 1. *Time - heart rate plot of the data prepared as a result of 3 hours of measurement.*

2.2 Data Pre-Processing

During the data collection process, several outliers were identified, likely resulting from sensor measurement errors. To enhance the quality and reliability of the dataset, these outliers were addressed by replacing them with the average of several preceding and succeeding data points. This technique, known as local averaging, effectively smooths the data and mitigates the impact of random measurement errors on the overall analysis. By using the average of surrounding values, preserved the continuity of the data, avoiding the introduction of arbitrary values that could skew the results. Additionally, local averaging helps reduce noise, dampening the influence of extreme values that could distort trends within the dataset. Ultimately, cleaning the dataset by replacing outliers ensures that the model is trained on more representative data, leading to improved predictions and generalization when applied to unseen data. This approach enhances the robustness of the predictive model by ensuring that the underlying trends in the heart rate data are accurately represented, thereby contributing to more reliable outcomes in the analysis.

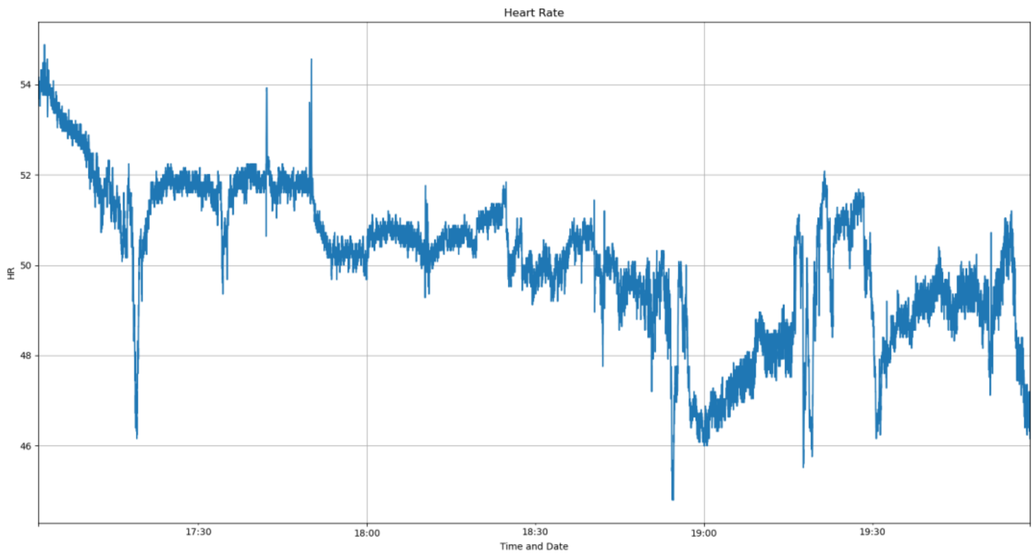

Figure 2. *Data after outliers are removed.*

In time series analysis, an autocorrelation graph is utilized to examine the degree to which current values in a dataset are related to their past values over different time lags. Autocorrelation measures the correlation of a time series with its own past values, helping to identify patterns, trends, and the presence of seasonality in the data. By plotting the autocorrelation function (ACF) against various lag values, researchers can discern whether past observations significantly influence future observations, which is essential for understanding the underlying structure of the data.

In this study, an autocorrelation graph was constructed to investigate the relationship between pulse values over time. Specifically, with a lag value of 180 seconds, the autocorrelation coefficient was calculated to be 75%. This substantial correlation suggests a strong relationship between pulse values at this lag, indicating that changes in pulse readings are significantly influenced by preceding values. Such a correlation value is compelling for time series analysis, as it implies the potential for predictive modeling based on historical data.

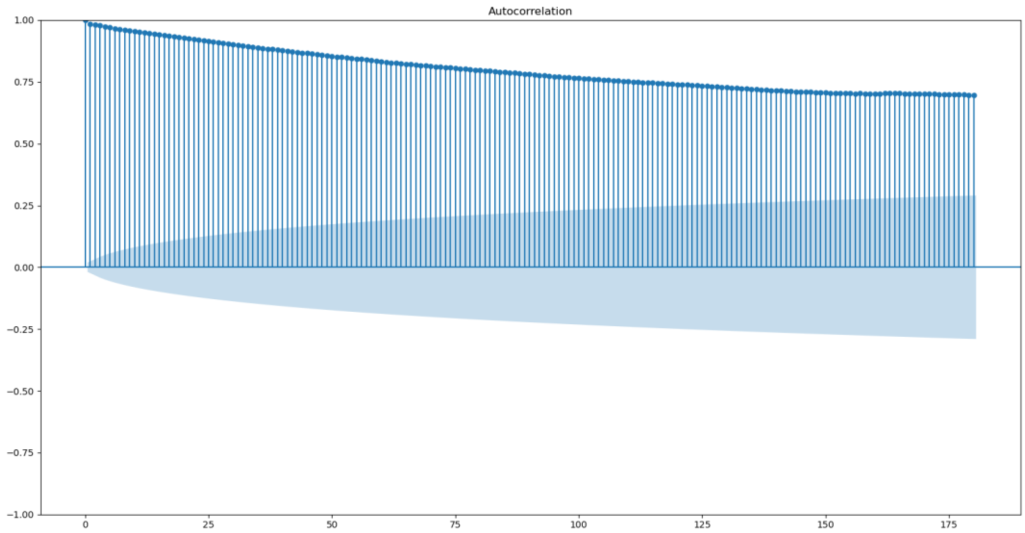

Figure 3. *Autocorrelation plot for lag = 180.*

A lag plot is a graphical tool used in time series analysis to visualize the relationship between observations at different time lags. By plotting the values of a time series against their lag values, it is possible to assess whether a linear or nonlinear relationship exists. In this study, lag plots were created for both lag values of 2 and 180 seconds. A clear linear trend was observed in both plots, indicating that the current pulse values are closely related to their past values at these particular lags. This linearity indicates that the data exhibits a persistent structure over time, reinforcing previous findings of nonstationarity. The presence of a linear trend in the lag plots serves as a secondary indicator of the nonstationarity of the data set, alongside the gradual decrease in the autocorrelation plot, since stationary data usually exhibit a more random distribution without a clear directional trend.

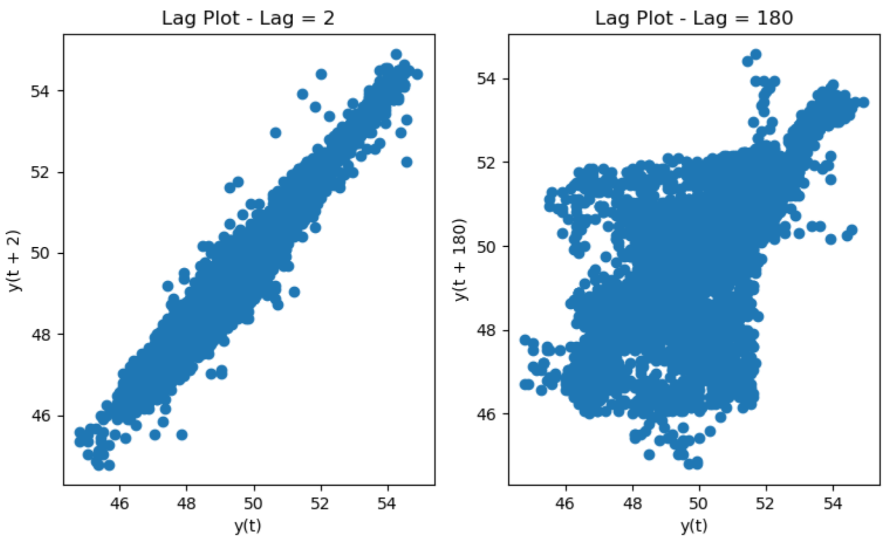

Figure 4. *Lag plot of heart rate values for different lag values.*

The statistical properties of stationary data remain constant over time, which means that key characteristics such as mean, variance, and autocorrelation do not exhibit changes as time progresses. Stationarity is a fundamental assumption in time series analysis, as many statistical methods and models rely on this condition for their validity. When working with non-stationary data, statistical inference can yield misleading results, leading to incorrect conclusions about the underlying processes.

To transform non-stationary data into stationary data, various techniques can be employed [15], with differencing being one of the most commonly used methods. This process involves subtracting the previous observation from the current observation, effectively removing trends and seasonality from the data.

$$Y_t = X_t - X_{t-1} \tag{1}$$

The differencing method was applied during the data preprocessing phase to achieve stationarity.

Figures 5 and 6 show lag autocorrelation plots after the data are made stationary by differencing.

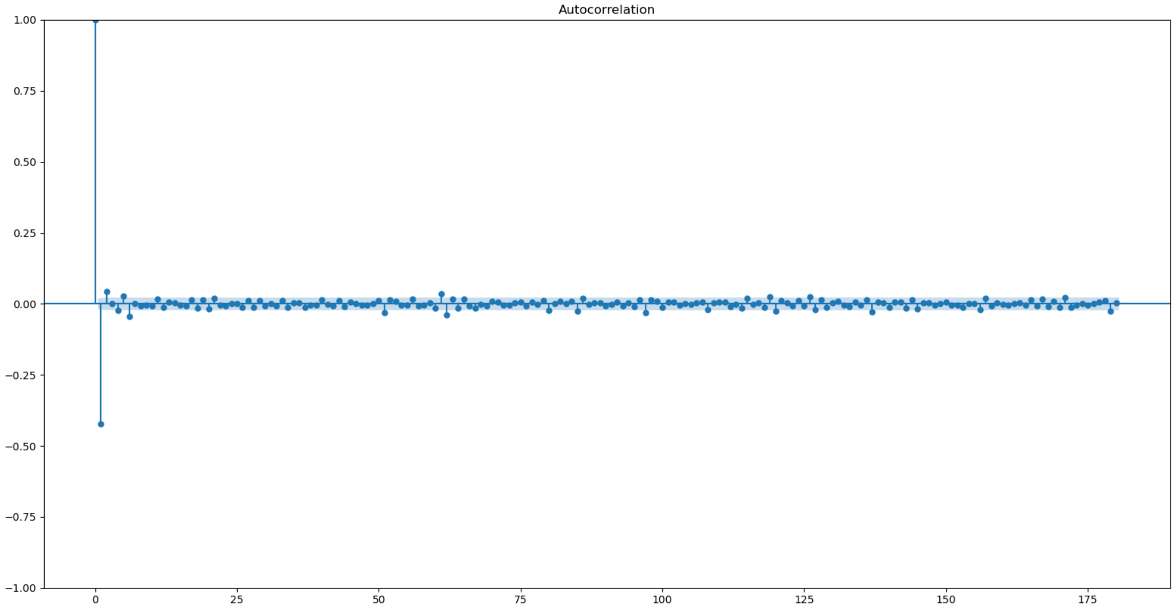

Figure 5. *Autocorrelation plot of stationary data.*

A rapid decrease observed in the autocorrelation graph reveals that the data set has a constant mean and variance, and this situation allows for more reliable results to be obtained in the modeling processes.

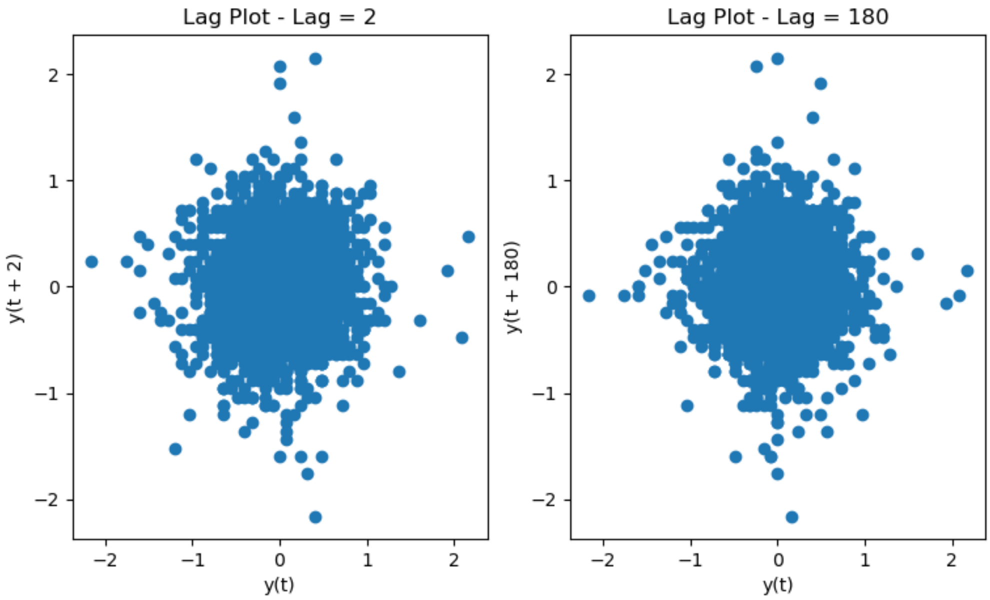

Figure 6. *Lag plot of stationary heart rate values for different lag values.*

The analyses performed on the autocorrelation graph and lag plot obtained after the differencing process show that the data has been made stationary. The random distribution of the lag plot shows that the dependency in the time series data has disappeared and the observations have become independent of each other.

After differencing, heart rate data were scaled to the range [-1, 1] using the “minmax scaler” . Observation windows were defined so that time series data could be processed with the deep learning models. Observation window is a vector representing the data observed during a certain time period.

$$X_t = (x_{t-1}, x_{t-2}, \dots, x_{t-n}) \quad (2)$$

In this context, the model utilizes these historical data points to predict the heart rate measurement at t+300 seconds, which represents a 5-minute prediction horizon. By leveraging the temporal dependencies captured in the observation windows, the deep learning models can effectively forecast future heart rate values. This capability is crucial for providing timely insights in real-time healthcare applications, where swift decision-making can significantly impact patient outcomes.

Table 1. *180-seconds observation and 300-seconds target Windows.*

Time (hours)	t-180	t-179	t-178	t-177	…	t+296	t+297	t+298	t+299
17:04:29	0.08	0.00	0.08	-0.08	…	-0.24	0.24	0.08	-0.24
17:04:30	0.00	0.08	-0.08	-0.08	…	0.24	0.08	-0.24	0.24
17:04:31	0.08	-0.08	-0.08	0.32	…	0.08	-0.24	0.24	-0.08
…	…	…	…	…	…	…	…	…	…
19:53:03	-0.32	-0.56	0.80	-0.16	…	0.16	-0.72	0.16	0.24
19:53:04	-0.56	0.80	-0.16	-0.80	…	-0.72	0.16	0.64	-0.24
19:53:05	0.80	-0.16	-0.44	0.06	…	-0.32	0.16	-0.72	0.64

As shown in Table 1, after being subjected to differencing, the data scaled to the range [-1,1] were shifted and prepared as 180-second observation (X) and 300-second target (Y) windows. As a result of these operations, the data preprocessing process was completed.

2.3. Data Splitting

As this study involves time series data, the dataset is split in chronological order**,** without shuffling to preserve the temporal structure. Specifically:

- 75% of the data is used for training,
- 10% for validation,
- 15% for testing.

This method ensures that the model is evaluated on unseen future data, which mimics real-world forecasting scenarios. The training **set** is used to update the model's parameters, while the validation set is reserved for assessing overfitting tendencies and for techniques like early stopping. Finally, the **test set** is used exclusively to evaluate the model's performance on data that was not involved in any part of the training process.

2.4. Training and Optimization Of the Models

To comprehensively evaluate the performance of various deep learning architectures in processing the prepared pulse rate data, developed models using Gated Recurrent Unit (GRU), Bidirectional Long Short-Term Memory (Bi-LSTM), Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) architectures. Each model was configured with varying depths, specifically utilizing 1 to 3 layers, to determine the optimal structure for data analysis.

To fine-tune the hyperparameters of these models, employed the grid search method [16], focusing on two critical parameters: the number of neurons within each layer and the dropout rate. The grid search configuration was set up as follows:

- The number of neurons in each layer was systematically varied from 10 to 200, increasing in increments of 10.
- The dropout rate was adjusted between 0.1 and 0.5, with increments of 0.1.

The use of Grid Search allowed us to comprehensively explore the hyperparameter space, facilitating the identification of the most effective configurations for each architecture. For the training of the models, employed the Adam optimizer, known for its efficiency and ability to adaptively adjust the learning rate. The models were trained for 50 epochs with a patience level of 5 for early stopping, preventing overfitting by monitoring the validation loss. Setted the learning rate to 0.0001 and used a batch size of 1, which enabled the model to update weights more frequently and respond dynamically to the training data.

3. Results

In the evaluation of the performance of each model, utilized the root mean square error (RMSE) metric, defined mathematically as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

RMSE was chosen as the primary evaluation metric for several reasons. Firstly, it provides a clear measure of the average magnitude of the prediction errors, allowing for a direct interpretation of how well the model's predictions align with the actual values. By squaring the errors, RMSE emphasizes larger discrepancies, making it particularly sensitive to outliers, which is important in healthcare applications where accurate pulse rate measurements are critical.

Secondly, RMSE has the same units as the target variable (in this case, heart rate), which facilitates intuitive understanding of the model's performance. This characteristic makes it easier for practitioners and stakeholders to assess the clinical relevance of the model's predictions.

The performance of each model was assessed based on the root mean square error (RMSE) for a lag of 3 minutes and a forecast horizon of 5 minutes. The results, summarized in Table 2, show that there are some variations in the predictive capabilities of the different architectures and configurations employed in this study.

Table 2. *The result of models for lag of 3 minutes and forecasting 5 minutes.*

Model	Number of Layers	Number of Neurons	Dropout Rate	RMSE
RNN	1	100	0.3	2.121
	2	[40,80]	[0.4,0.2]	2.041
	3	[40,120,80]	[0.5,0.3,0.5]	1.971
LSTM	1	70	0.5	2.071
	2	[120,100]	[0.4,0.3]	1.881
	3	[160,120,40]	[0.3,0.5,0.3]	1.839
GRU	1	150	0.2	2.008
	2	[140,50]	[0.5,0.5]	2.045
	3	[70,120,70]	[0.3,0.5,0.3]	1.940
Bi-LSTM	1	70	0.2	2.035
	2	[70,120]	[0.5,0.5]	2.173
	3	[120,140,110]	[0.3,0.5,0.3]	2.223

The results show that the RNN model with three layers achieved the best performance among all configurations, with an RMSE of 1.971, while the one-layer RNN had the worst performance with an RMSE of 2.121. For the LSTM model, the three-layer configuration also performed best, resulting in an RMSE of 1.839, whereas the two-layer LSTM produced the highest RMSE of 2.071. In the case of the GRU model, the two-layer setup yielded the best performance with an RMSE of 1.940, while the three-layer configuration had the least favorable outcome, resulting in an RMSE of 2.045. The Bi-LSTM model demonstrated the highest performance with a one-layer setup, achieving an RMSE of 2.035, while the three-layer configuration recorded the highest RMSE of 2.223.

4. Conclusions

In this study, comprehensively evaluated the performance of various deep learning architectures (i.e. Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), Gated Recurrent Unit (GRU), and Recurrent Neural Networks (RNN)) in predicting heart rate data by collecting 10,597 pulse values for three hours. Findings revealed that the LSTM model configured with three layers demonstrated superior predictive capabilities, achieving the lowest Root Mean Square Error (RMSE) of 1.839. This result not only underscores the effectiveness of LSTM networks in handling time series data but also highlights their potential to significantly enhance the accuracy of heart rate predictions, which is critical for the development of advanced e-health monitoring systems.

The implications of these findings are far-reaching, as accurate heart rate predictions can lead to timely interventions and improved patient outcomes, particularly in remote monitoring scenarios where immediate clinical assessment may not be possible. The integration of reliable predictive models in e-health systems could enable healthcare providers to make informed decisions swiftly, thereby optimizing patient care and resource allocation.

However, it is important to acknowledge certain limitations within this study. The dataset utilized was confined to a specific population and time duration, which may limit the generalizability of the results across diverse demographic groups and clinical conditions. Future research could address this limitation by incorporating a broader range of datasets that include various age groups, health statuses, and environmental conditions. Additionally, the integration of complementary physiological signals—such as blood pressure, oxygen saturation, and even physical activity levels—could further enhance the predictive power of these models, providing a more holistic view of patient health.

Declaration of Interest

The authors declare that there is no conflict of interest.

References

[1] World Health Organization, "Cardiovascular Diseases (CVDs)," Available: https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).

[2] H. Rahman, M. U. Ahmed, and S. Begum, "Vision-based remote heart rate variability monitoring using camera," in *Internet of Things (IoT) Technologies for HealthCare: 4th Int. Conf., HealthyIoT 2017, Angers, France, Oct. 24-25, 2017, Proc.*, 2018, pp. 10–18.

[3] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, and J. Dean, "A guide to deep learning in healthcare," *Nat. Med.*, vol. 25, no. 1, pp. 24–29, 2019.

[4] M. Oyeleye, T. Chen, S. Titarenko, and G. Antoniou, "A predictive analysis of heart rates using machine learning techniques," *Int. J. Environ. Res. Public Health*, vol. 19, no. 4, pp. 2417, 2022.

[5] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "M5 accuracy competition: Results, findings, and conclusions," *Int. J. Forecast.*, vol. 38, no. 4, pp. 1346–1364, 2022.

[6] R. J. Hyndman, *Forecasting: Principles and Practice*. OTexts, 2018.

[7] R. B. Govindan, A. N. Massaro, N. Niforatos, and A. Du Plessis, "Mitigating the effect of non-stationarity in spectral analysis—An application to neonate heart rate analysis," *Comput. Biol. Med.*, vol. 43, no. 12, pp. 2001–2006, 2013.

[8] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. W. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, 2017, pp. 2627–2633.

[9] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process.*, 2014, pp. 1724–1734.

[10] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv* preprint arXiv:1409.0473, 2014.

[11] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "Geoman: Multi-level attention multi-order for geo-sensory time series prediction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, 2018, pp. 3428–3434.

[12] H. Lin, S. Zhang, Q. Li, Y. Li, J. Li, and Y. Yang, "A new method for heart rate prediction based on LSTM-BiLSTM-Att," *Measurement*, vol. 207, p. 112384, 2023.

[13] H. Ni et al., "Time Series Modeling for Heart Rate Prediction: From ARIMA to Transformers," *arXiv* preprint arXiv:2406.12199, 2024.

[14] A. Staffini, T. Svensson, U. I. Chung, and A. K. Svensson, "Heart rate modeling and prediction using autoregressive models and deep learning," *Sensors*, vol. 22, no. 1, p. 34, 2021.

[15] R. Salles, K. Belloze, F. Porto, P. H. Gonzalez, and E. Ogasawara, "Nonstationary time series transformation methods: An experimental review," *Knowl.-Based Syst.*, vol. 164, pp. 274–291, 2019.

[16] J. Bergstra and Y. Bengio, "Random Search for Hyper-Parameter Optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, 2012.