Predicting Medical Emergencies Using Machine Learning

Abstract

This project presents a machine learning-based solution to predict medical emergencies (like seizures or heart attacks) in real-time by analysing patient data, aiming to provide timely alerts for doctors. Comparative analysis of six models—LSTM, CNN, CNN+LSTM, MLP, VGG-16, and MobileNet—indicated that VGG-16 yielded the best results for the selected dataset due to the high similarity between predicted and original graphs. This report outlines the methodology, model performance, and potential real-world applications of this system.

1. Introduction

With advances in healthcare technology, real-time monitoring of vital signs has become crucial in predicting critical medical events. Predicting emergencies like heart attacks and seizures can offer life-saving lead time for intervention. This project focuses on developing a predictive model using machine learning to analyse heart rate data, identify patterns, and trigger alerts before emergencies occur.

The objectives of this project are:

- To compare and analyse different machine learning models for predicting medical emergencies.
- To determine the best model for accurately predicting emergencies in real-time.
- To implement a prototype that demonstrates the model's effectiveness in predicting emergencies based on heart rate data.

2. Data Collection and Preprocessing

Dataset Summary

The dataset contains heart rate readings across four variables (T1, T2, T3, T4) and comprises 1800 rows of data. These readings capture variations in heart rate over time, serving as a basis for predictive modeling.

Preprocessing Steps

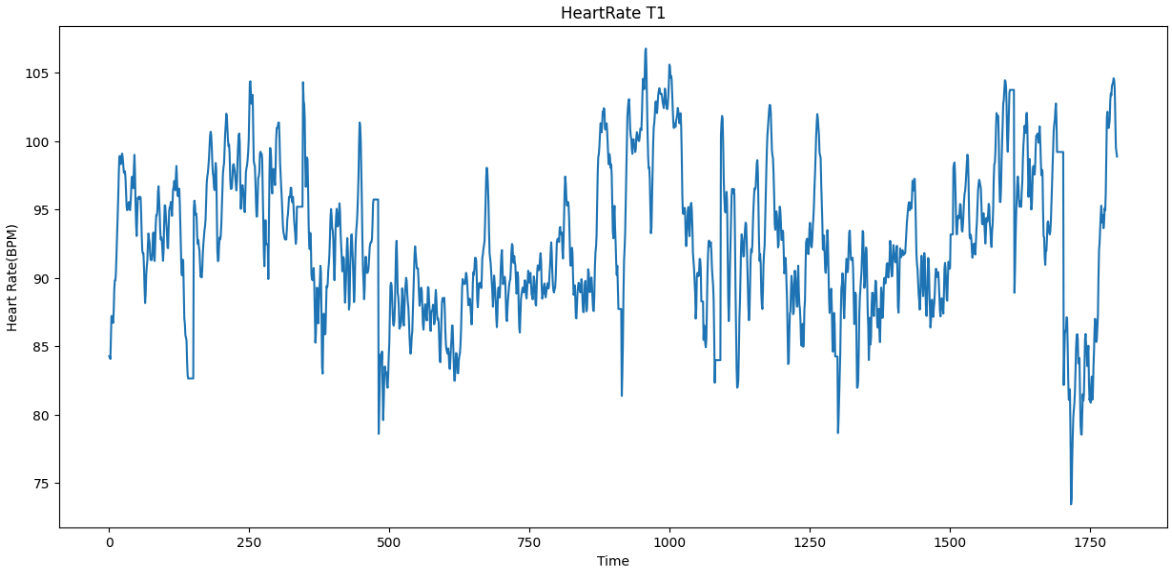
The preprocessing workflow included:

- **Standardisation and Normalisation**: Scaling data to ensure uniformity across inputs.
- **Time-Series Transformation**: Converting raw data into time-lagged sequences suitable for supervised learning. A custom `series_to_supervised` function was

developed, generating inputs based on a sliding time window of 29 steps to improve temporal pattern recognition.

- **Data Splitting**: Partitioning the data into training and testing sets with an 80-20 split, ensuring sufficient data for model validation.

The figure below provides an example of the time-lagged data transformation.



3. Model Selection

Model Comparison
We evaluated six models, each chosen for its unique approach to handling sequential and high-dimensional data:

- **LSTM (Long Short-Term Memory)**: Known for capturing long-term dependencies in sequential data.
- **CNN (Convolutional Neural Network)**: Effective in identifying local patterns in time-series data.
- **CNN+LSTM**: Combines CNN's feature extraction with LSTM's sequence processing.
- **MLP (Multi-Layer Perceptron)**: A simpler model serving as a baseline.
- **VGG-16**: A deep convolutional network adapted for time-series prediction, leveraging its pre-trained weights for improved pattern detection.
- **MobileNet**: Known for its efficiency and effectiveness in handling real-time applications.

The hyperparameters for each model were tuned based on initial experimentation, with VGG-16 configured to optimize the real-time prediction task for high-dimensional, time-sequenced data.

4. Experimental Setup and Evaluation Metrics

Model Training
Each model was trained with an epoch count of 100, batch size of 256, and an Adam optimizer set with a learning rate of 0.0001. Training was conducted on the training subset of data, and model validation was performed using the testing subset.

Evaluation Metrics
To assess each model's performance, the following metrics were utilized:
Root Mean Squared Error (RMSE): To measure prediction error.
R² Score: To indicate the proportion of variance explained by the model.
Correlation Coefficient: To assess the similarity between predicted and actual values.
Mean Absolute Percentage Error (MAPE): For percentage-based error analysis.

These metrics were selected for their ability to provide insights into both error magnitude and predictive accuracy, crucial for real-time applications.

Figure 1: MLP Performance Graphs

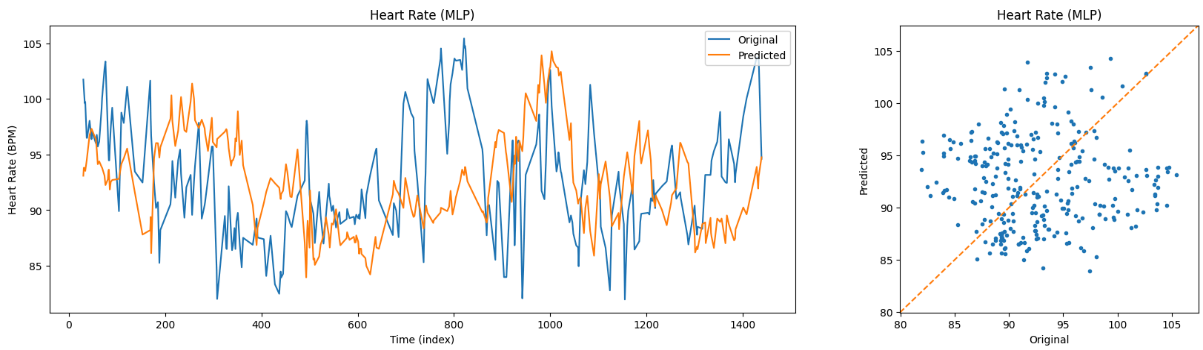


Table 1: MLP SCORES

R2	Corr	RMSE	MAPE
-65.443	0.029	6.729	6.099

Figure 2: CNN Performance Graphs

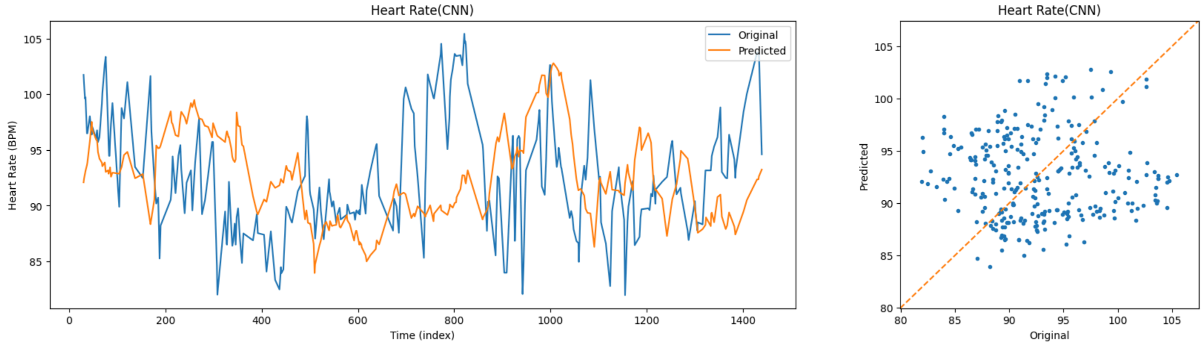


Table 2: CNN SCORES

R2	Corr	RMSE	MAPE
-59.388	-0.001	6.605	5.901

Figure 3: LSTM Performance Graphs

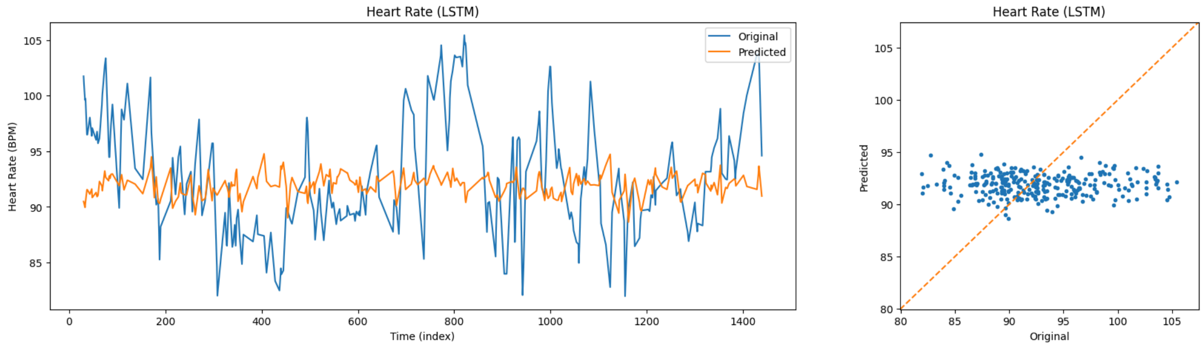


Table 3: LSTM SCORES

R2	Corr	RMSE	MAPE
0.16	0.122	5.245	4.567

Figure 4: CNN+LSTM Performance Graphs

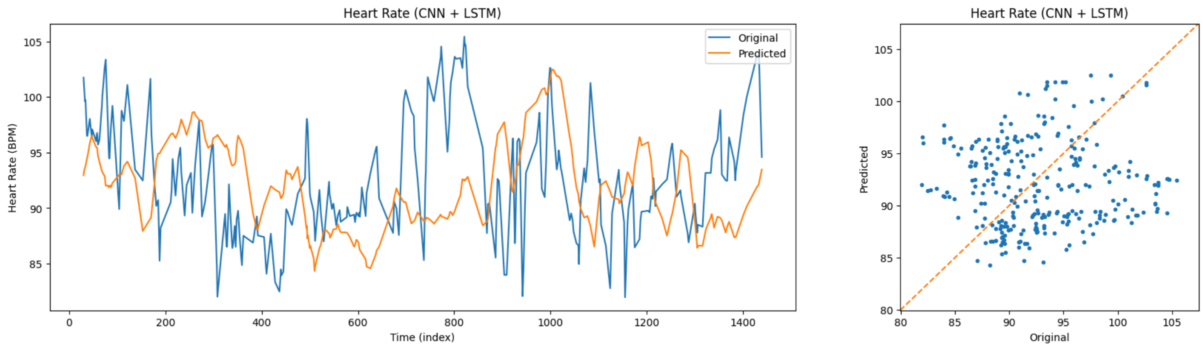


Table 4: CNN+LSTM SCORES

R2	Corr	RMSE	MAPE
-60.416	0.012	6.626	5.965

Figure 5: VGG-16 Performance Graphs

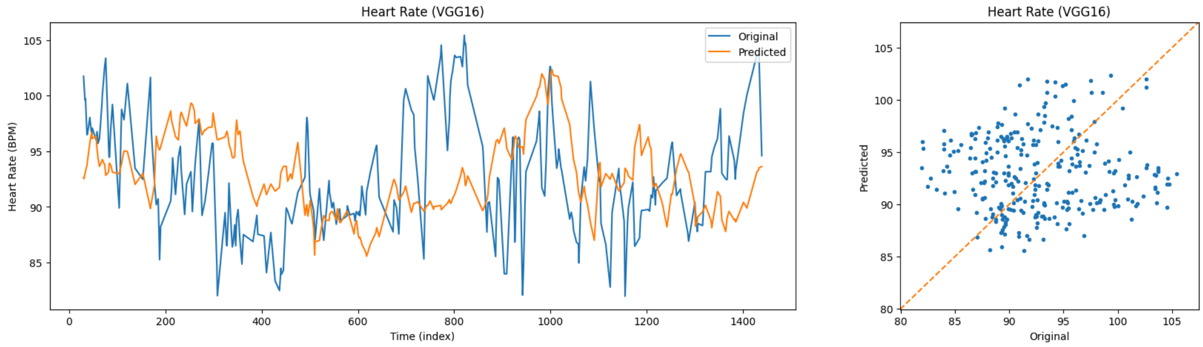


Table 5: VGG-16 SCORES

R2	Corr	RMSE	MAPE
-48.367	0.005	6.373	5.67

Figure 6: MOBILENET Performance Graphs

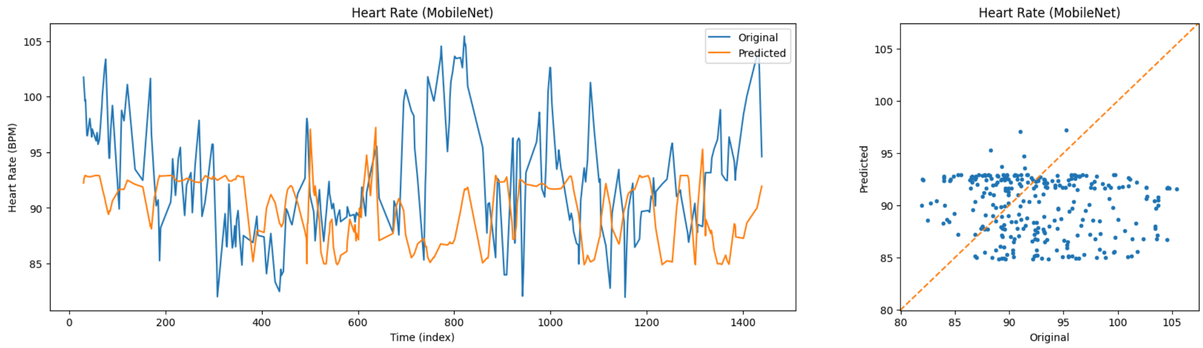
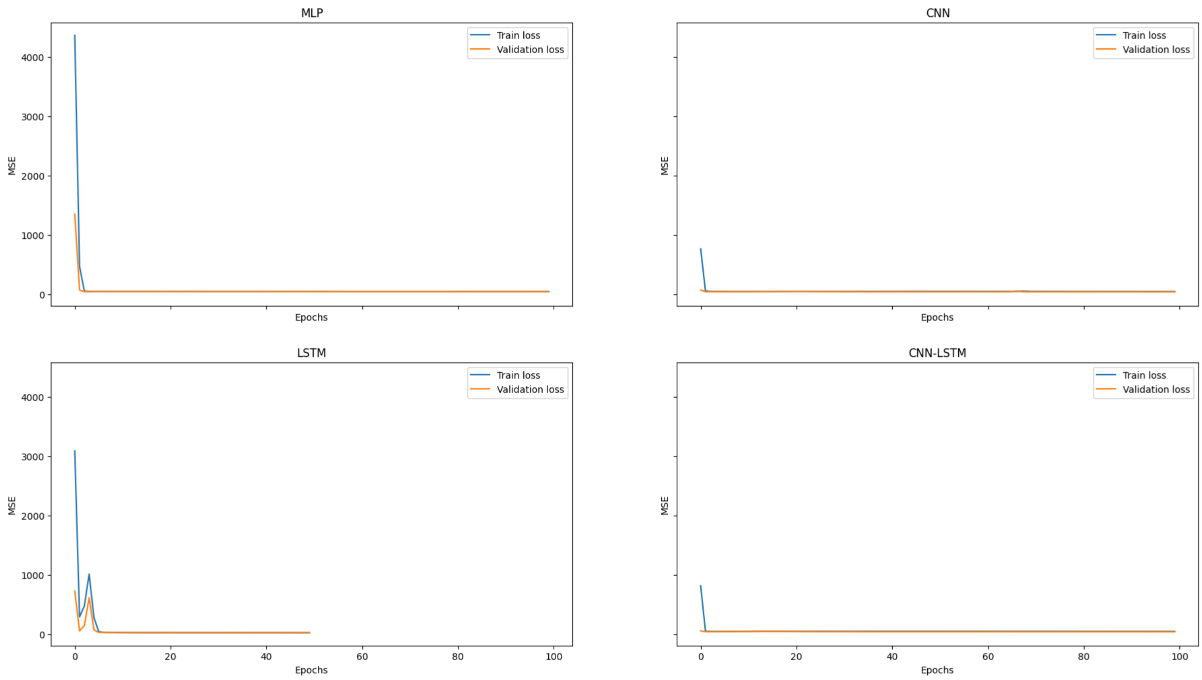
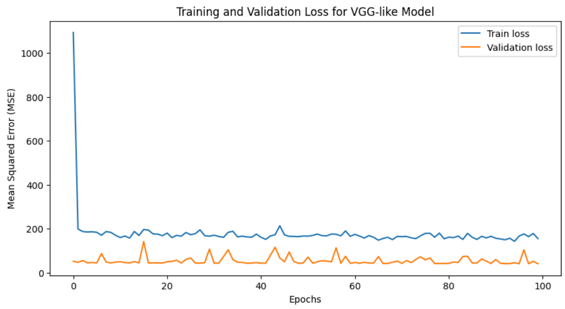


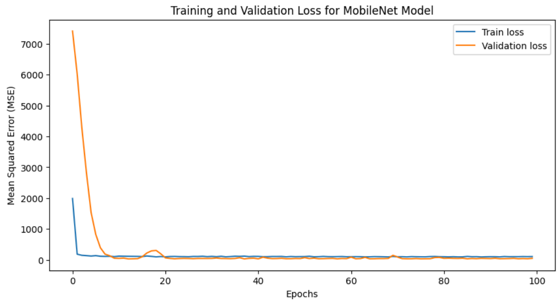
Table 6: MOBILENET SCORES

R2	Corr	RMSE	MAPE
-72.289	-0.124	5.519	5.824

Figure 7: Train Loss and Validation Loss Graphs



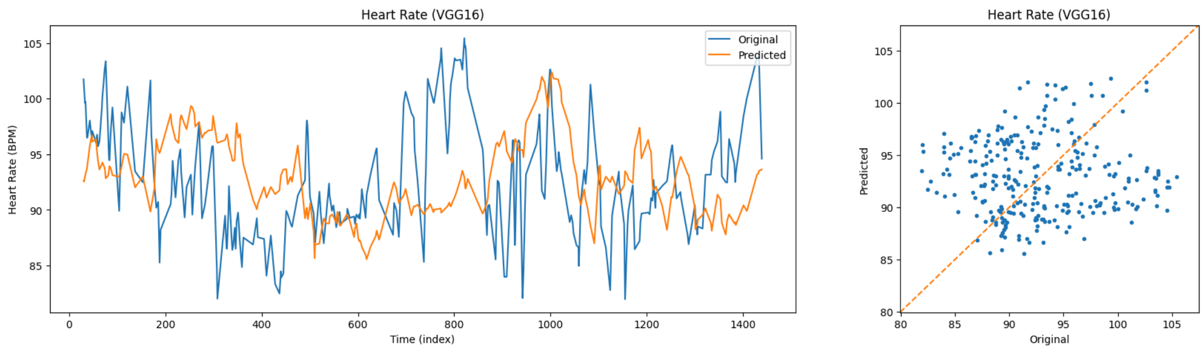




5. Results

After comparative analysis, VGG-16 demonstrated superior performance, showing the highest alignment between predicted and original values. The table below summarizes the results, and the following graphs illustrate the similarity between predicted and actual graphs, validating VGG-16 as the optimal model for this dataset.

Graph 1: VGG-16 Predicted vs Actual Graph



The primary finding is that VGG-16, with its deeper architecture, effectively captures complex temporal patterns, resulting in high predictive accuracy. The CNN+LSTM model also performed well but did not match VGG-16's precision in aligning with original data trends.

6. Discussion

Model Insights
VGG-16's performance can be attributed to its robust architecture, which captures intricate patterns in sequential heart rate data. The model's pre-trained convolutional layers proved effective in extracting spatial and temporal features, making it well-suited to this type of predictive task.

Limitations and Challenges
Data Size: The dataset was relatively small, and a larger dataset could further improve model accuracy.
Real-Time Constraints: The models need to be optimized for real-time deployment, especially for resource-constrained environments.

Future Improvements
Integrating additional physiological features (like blood pressure and oxygen saturation) could further enhance prediction accuracy. Additionally, testing the model across more varied datasets would improve its generalizability.

7. Conclusion

This project demonstrated the feasibility of using machine learning to predict medical emergencies based on real-time heart rate data. Through comparative analysis, VGG-16 emerged as the most effective model, with its predictions closely matching actual trends. This solution has significant potential for real-world application in healthcare settings, where timely alerts can be life-saving.

References

1. Heart Rate Dataset
2. Relevant machine learning and medical literature