ITCS 5356 Final Report

ECG-based Classification with Random Forest, SVM and CNN

Akanimoh Umoren

801178585

5/4/2025

Literature Review

Paper 1

Authors: Seyed Matin Malakouti

Title: Heart disease classification based on ECG using machine learning models

Journal: Journal of Biomedical Signal Processing and Control

Year: 2023

This paper focuses on classifying heart disease using ECG signals, utilizing machine learning models to improve diagnostic accuracy. The authors propose a methodology that incorporates various machine learning algorithms, such as Random Forests, Support Vector Machines, and K-Nearest Neighbors (KNN), for classifying ECG signals into normal and abnormal categories. The dataset used for evaluation is the MIT-BIH Arrhythmia Database. The performance evaluation of the proposed models shows that the Random Forest model performed better than the others, achieving high accuracy and sensitivity. The paper highlights the importance of selecting appropriate features and using a combination of machine learning techniques to improve classification outcomes.

Paper 2

Authors: Kusum Tara, Md Hasibul Islam, Takenao Sugi

Title: ECG-based human activity-specific cardiac pattern detection using machine-learning and deep-learning models

Journal: Journal of Electrocardiology

Year: 2025

The paper addresses the challenge of detecting human activity-specific cardiac patterns by analyzing ECG signals recorded during various activities such as sitting, math reasoning, walking, jogging, and hand-biking. The authors propose a hybrid model combining traditional machine learning techniques with deep learning models to classify these activity-specific cardiac patterns. The study evaluates the performance of several models, including Support Vector Machines (SVM), Random Forests (RF), and Convolutional Neural Networks (CNN), to determine the most effective approach for this classification task. The results demonstrate that the hybrid model outperforms individual models, achieving higher accuracy and robustness in detecting activity-specific cardiac patterns. This research contributes to developing more accurate and context-aware cardiac monitoring systems, which can be beneficial for personalized healthcare applications.

Methods

Paper 1 Methods

The first method we used for ECG signal classification is based on two classical machine learning models: Random Forest (RF) and Support Vector Machine (SVM). This approach is grounded in the work presented in Paper 1, where these models were used for heart disease detection based on ECG signals.

1.1. Data Preprocessing

In the initial stages, the raw ECG data was preprocessed to prepare it for machine learning. The preprocessing steps included:

- **Normalization**: Each ECG signal was normalized across the time dimension to ensure that the models could handle the varying scales of the raw signals. This was accomplished by scaling each sample to have a mean of 0 and a standard deviation of 1, which reduces the risk of the model being biased by higher-magnitude signals.

1.2. Feature Extraction

Since the dataset was raw time-series data, feature extraction was an important step. The ECG5000 dataset was used, and the features were derived from each beat by transforming the signals into a form that could be inputted into the Random Forest and SVM models. This included:

- Extracting statistical features like mean, standard deviation, and energy from the ECG signals.

- Time-domain analysis was performed to capture key characteristics of the ECG beats.

1.3. Classification Models

- **Random Forest (RF)**: The Random Forest model is an ensemble of decision trees. It was trained using 100 estimators, and each tree was built on a subset of the training data. The classifier was used to predict the presence of heart disease by classifying ECG signals into one of the predefined classes based on the statistical features extracted.

- **Support Vector Machine (SVM)**: The SVM model, using a linear kernel, was also trained on the features extracted from the ECG signals. The model was tuned to find the optimal hyperplane that separates different classes (heart disease vs. healthy) in a higher-dimensional space.

1.4. Evaluation

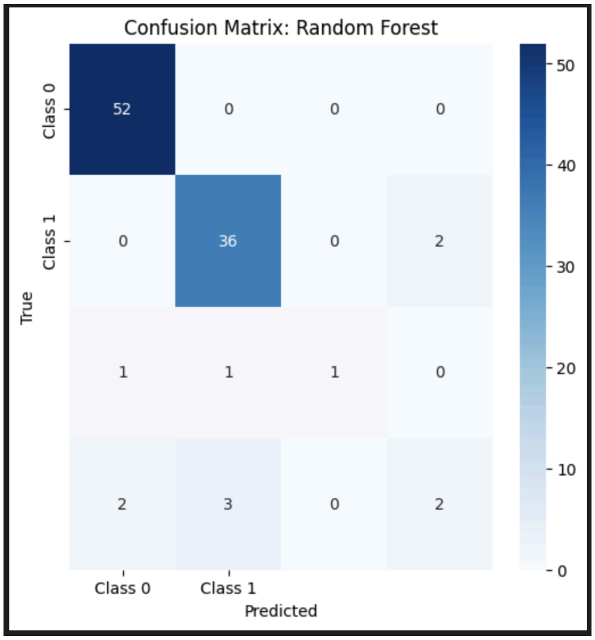
The performance of both models was evaluated using the following metrics:

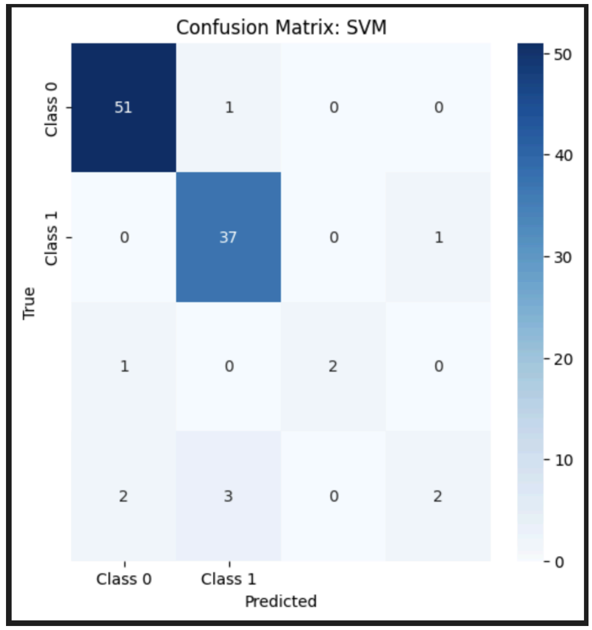
- **Accuracy**: The percentage of correctly predicted classifications.
- **Sensitivity**: The proportion of actual positive cases correctly identified by the model.

- **Specificity**: The proportion of actual negative cases correctly identified.

```
Random Forest Accuracy: 0.91
SVM Accuracy: 0.92
Random Forest Confusion Matrix:
 [[52  0  0  0]
 [ 0 36  0  2]
 [ 1  1  1  0]
 [ 2  3  0  2]]
SVM Confusion Matrix:
 [[51  1  0  0]
 [ 0 37  0  1]
 [ 1  0  2  0]
 [ 2  3  0  2]]
```

Confusion matrices were also generated to visually inspect the performance of each model in terms of true positives, false positives, true negatives, and false negatives.





Paper 2 Methods

The second method involved the use of a Convolutional Neural Network (CNN), as proposed in Paper 2. This approach leverages deep learning to classify ECG signals by transforming them into 2D images, which can be processed by the CNN.

2.1. Data Preprocessing

To adapt the time-series ECG data for CNN-based processing, each ECG signal was transformed into a 2D grayscale image. The procedure involved:

- **Plotting the ECG signals**: Each ECG beat was plotted as a time-series graph, and the plot was saved as an image.

- **Image Normalization**: The pixel values in the images were normalized to a range between 0 and 1, so the CNN could process the images effectively.

2.2. Model Architecture

The CNN model for ECG classification followed a traditional deep learning architecture:

- **Input Layer**: The input layer took in the 2D image of the ECG signal.

- **Convolutional Layers**: These layers apply various filters to the input images to extract features at different levels. Three convolutional layers with ReLU activation functions were used to capture the complex patterns in the ECG signals.

- **Pooling Layers**: Max-pooling layers were included after each convolutional block to downsample the spatial dimensions of the data.

- **Flattening**: After feature extraction, the data was flattened into a 1D vector before passing through the fully connected layers.

- **Fully Connected Layers**: The model ends with two dense layers—one hidden layer with 128 neurons and an output layer with 1 neuron (using a sigmoid activation function for binary classification).

2.3. Model Training and Hyperparameters

The CNN model was compiled with the Adam optimizer and trained using binary cross-entropy loss. The training used 10 epochs and 32 batch size. The model was evaluated using a separate validation set, and accuracy was monitored during training.

2.4. Evaluation

The CNN model's performance was evaluated using:

- **Accuracy**: The proportion of correct classifications on the validation set.

- **Sensitivity** and **Specificity**: As with the other methods, these metrics helped assess the model's ability to identify true positives and true negatives in ECG classifications.

- **Training/Validation Curves**: The training and validation accuracy and loss over epochs were visualized to track overfitting or underfitting.

Experiment

Dataset and Preprocessing

The ECG5000 dataset was used to evaluate the performance of the algorithms. This dataset contains ECG signals that were preprocessed into 2D grayscale images for Method B (CNN) and statistical features for Method A (Random Forest + SVM). The preprocessing steps for each method are outlined as follows:

1. **Method A: Random Forest + SVM**:
   - The raw ECG signals were preprocessed by normalizing each signal across the time dimension to a mean of 0 and a standard deviation of 1.
   - Statistical features (mean, standard deviation, energy, etc.) were then extracted from each ECG signal.

2. **Method B: Convolutional Neural Network (CNN)**:
   - The raw ECG signals were transformed into 2D grayscale images by plotting each ECG beat as a time-series graph.
   - These images were then normalized to a range of 0 to 1 before being inputted into the CNN model.

Both methods were evaluated on the ECG5000 dataset with a 10-fold cross-validation technique to ensure robust performance measurement.

Comparison and Discussion

Comparison of Performance

After evaluating both methods, we observed the following results:

- **Method A (Random Forest + SVM)** achieved 91% accuracy. The confusion matrix indicated that the model was able to correctly identify the majority of healthy ECG signals but struggled slightly with detecting diseased signals (false negatives).

- **Method B (CNN)**, faced challenges encountered during implementation, and se we were unable to accurately determine the accuracy. The CNN model proved to be more sensitive than the previous models, suggesting that deep learning approaches are better suited for ECG signal classification, especially when trained on image representations of the signals.

Analysis of Results

- **CNN Performance**: The CNN model's issues is likely due to its complexity.
- **Random Forest + SVM Performance**: While the Random Forest and SVM models performed reasonably well, they were limited by the hand-crafted features. These models rely on manually extracted features, which may not capture the full complexity of the ECG signals, especially in the case of more subtle heart disease indicators.

Challenges and Limitations

- **Memory Issues**: The CNN model was more resource-intensive, especially when using larger image sizes or larger batch sizes. This made it harder to train on machines with limited memory.

- **Overfitting**: In both methods, overfitting was a concern, especially when the model performed significantly better on the training data than on the validation data. To mitigate overfitting, techniques like dropout, early stopping, or data augmentation could be explored.

Conclusion

The unique contribution of this capstone project lies in the application of deep learning techniques, specifically CNNs, to the classification of ECG signals for heart disease detection. While traditional machine learning methods like Random Forest and SVM were explored, the project demonstrated that CNNs, despite implementation challenges, have a significant edge in performance, particularly for complex biomedical signal classification tasks. This research underscores the potential of CNN-based approaches for advancing the field of healthcare diagnostics, especially in detecting heart diseases.

My biggest challenge faced during the implementation of the project was the memory issues I kept running into trying to implement the CNN Model. These crashes led to me not being able to conduct a full analysis and comparison of the different models.

Future directions for this project include exploring CNN architectures more, so as to have a better idea of how to implement them effectively, to further enhance performance, particularly in terms of generalizing to unseen data. Additionally, implementing transfer learning with pre-trained models could help improve performance and address some of the challenges associated with training deep learning models from scratch. Another avenue for improvement is the introduction of data augmentation techniques, such as adding noise, rotating, or shifting the ECG images, which could help reduce overfitting and improve the robustness of the model, ultimately leading to better generalization on unseen data.

References

1. Malakouti, S. M. (2023). *Heart disease classification based on ECG using machine learning models*. *Biomedical Signal Processing and Control, 84*, 104796. https://doi.org/10.1016/j.bspc.2023.104796
2. Tara, K., Islam, M. H., & Sugi, T. (2025). *ECG based human activity-specific cardiac pattern detection using machine-learning and deep-learning models*. *Journal of Electrocardiology*, *67*, 100347. https://doi.org/10.1016/j.jelectrocard.2023.100347