ECG Classification using Deep Learning

MIT-BIH Arrythmia Dataset

Lewis Richter

University West, Trollhättan, Sweden

GitHub: ECGClassification.git

UNIVERSITY WEST

Creating change together.

EU AI Regulations Impact

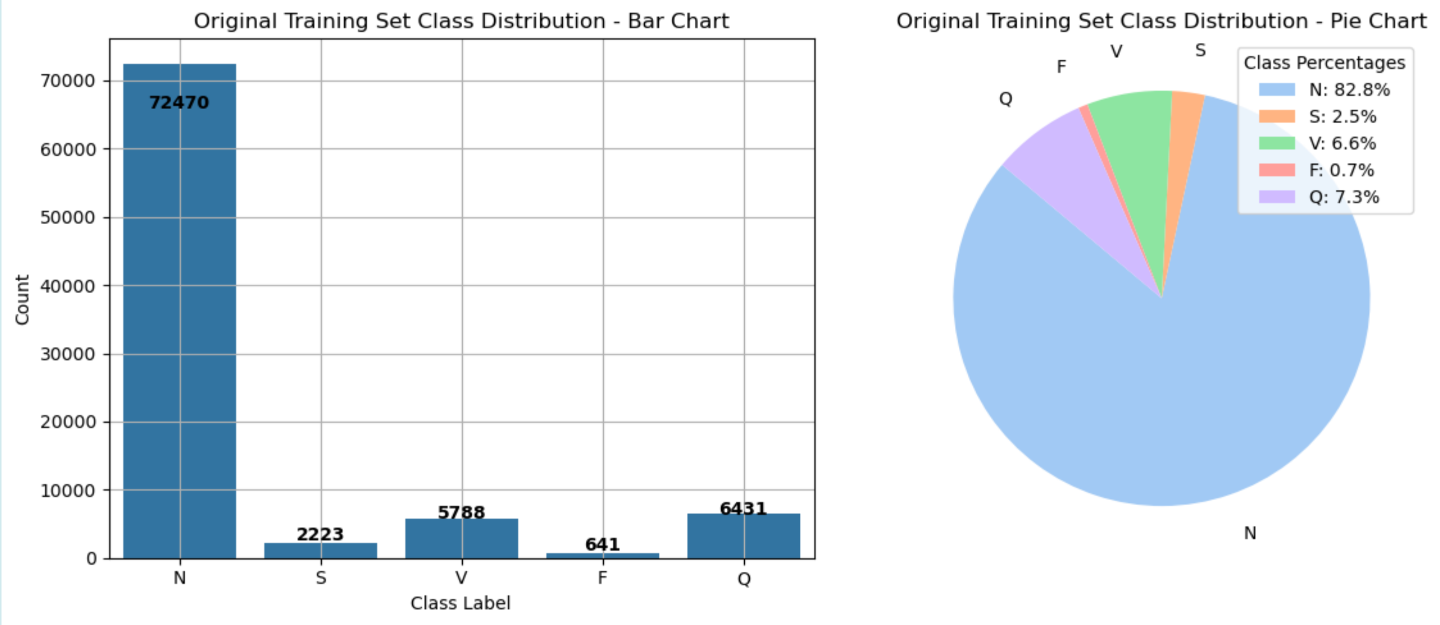
- Why it matters: Healthcare AI is high-risk under EU law.
- Implications:
  - Requires GDPR compliance
  - Mandatory risk assessment
  - Transparent design and human oversight
- Outcome: Your model must be explainable and medically safe before deployment.

Requirements and Specifications

- Main Goal: Classify ECG heartbeats into 5 categories
- Requirements:
    1. Dataset: MIT-BIH Arrhythmia Dataset
    2. Accuracy ≥ 95%
    3. Training Time ≤ 5 hours on M2 Mac Book Air
    4. Generalization on unseen data
    5. Evaluation with accuracy, precision, recall and F1-score

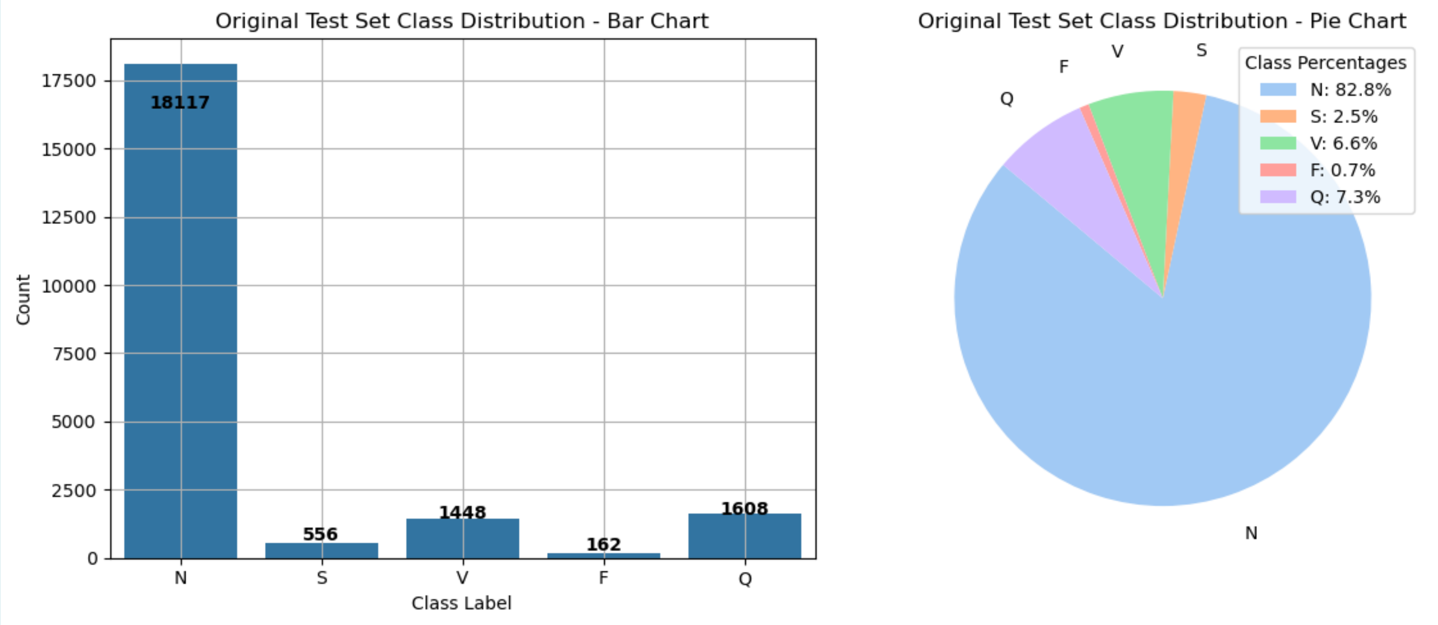
Dataset Overview

- Dataset: MIT-BIH Arrhythmia (via Kaggle)
- 5 Classes: Normal, Supraventricular, Ventricular, Fusion, Unknown
- Samples:
  - Training: 87,554 sample heartbeats
  - Test: 21,890 sample heartbeats

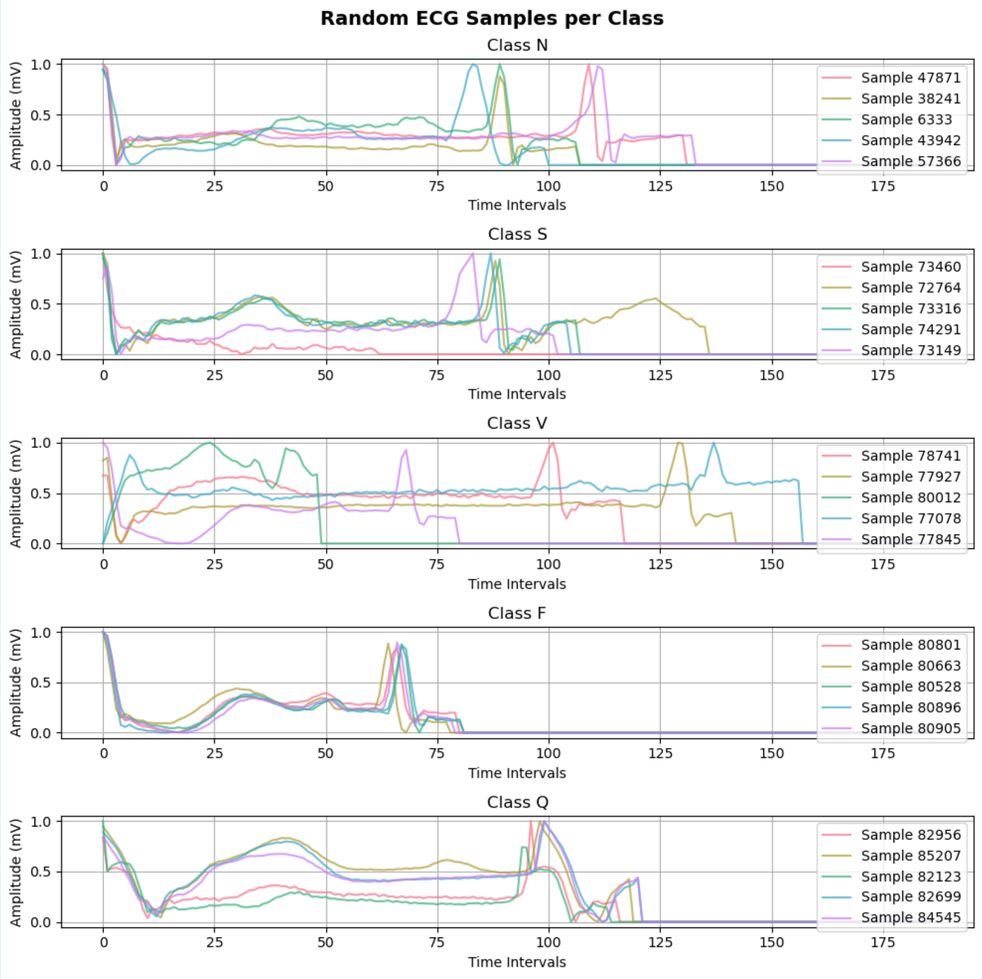


Dataset Overview

- Dataset: MIT-BIH Arrhythmia (via Kaggle)
- 5 Classes: Normal, Supraventricular, Ventricular, Fusion, Unknown
- Samples:
  - Training: 87,554 sample heartbeats
  - Test: 21,890 sample heartbeats



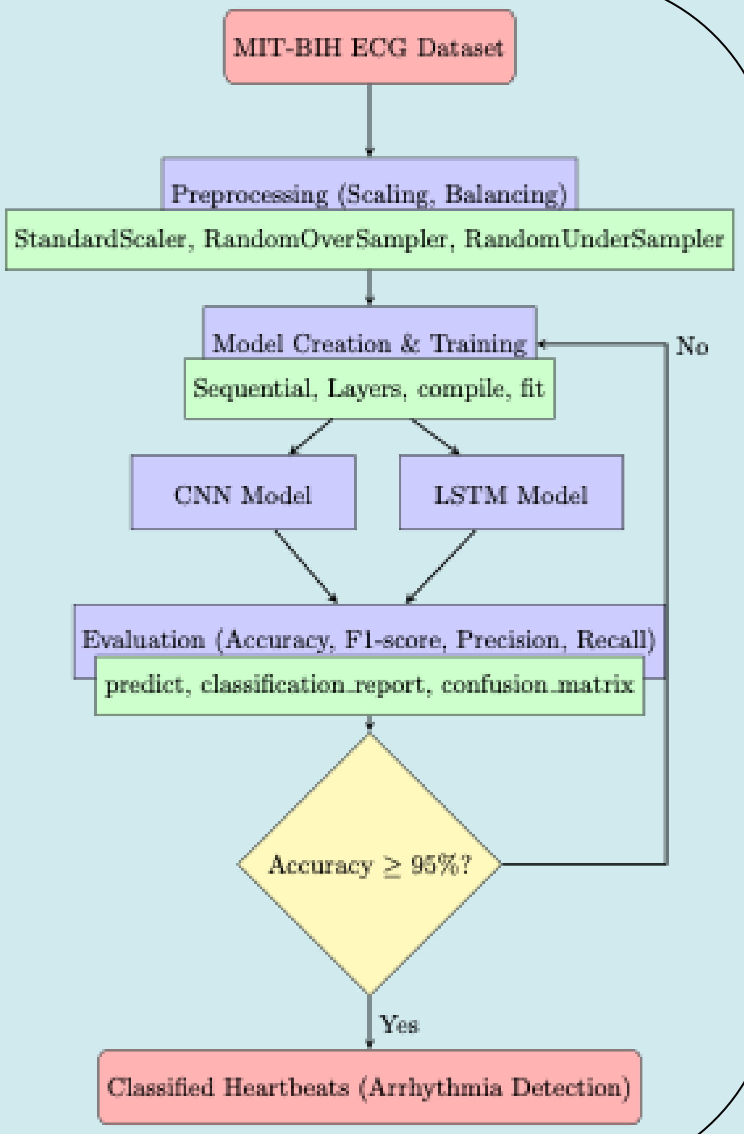
Dataset Overview



- 5 Classes:
  - Normal (N),
  - Supraventricular (S),
  - Ventricular (V),
  - Fusion (F),
  - Unknown (Q)
- Individual Beats extracted from 24 hours ECG
- Labelled independently by two experts

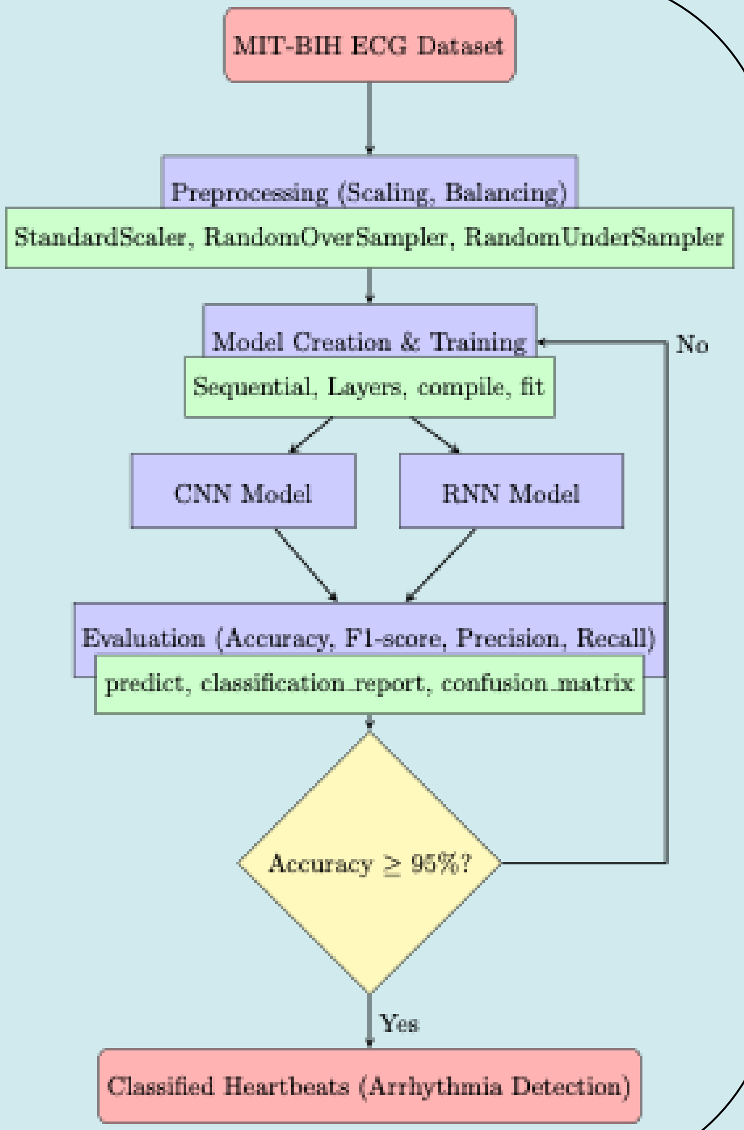
System Overview

- Dataset: MIT-BIH Arrhythmia (via Kaggle)
- Preprocessing:
    - Normalizing using StandardScaler:
        - Remove Mean
        - Scale to Unit Variance
    - Balancing using ImbLearn Library:
        - Random Under Sampling of Majority Class (N) to 50% (36,235 Heartbeats)
        - Random Over Sampling of Minority Classes to the same number of Heartbeats



System Overview

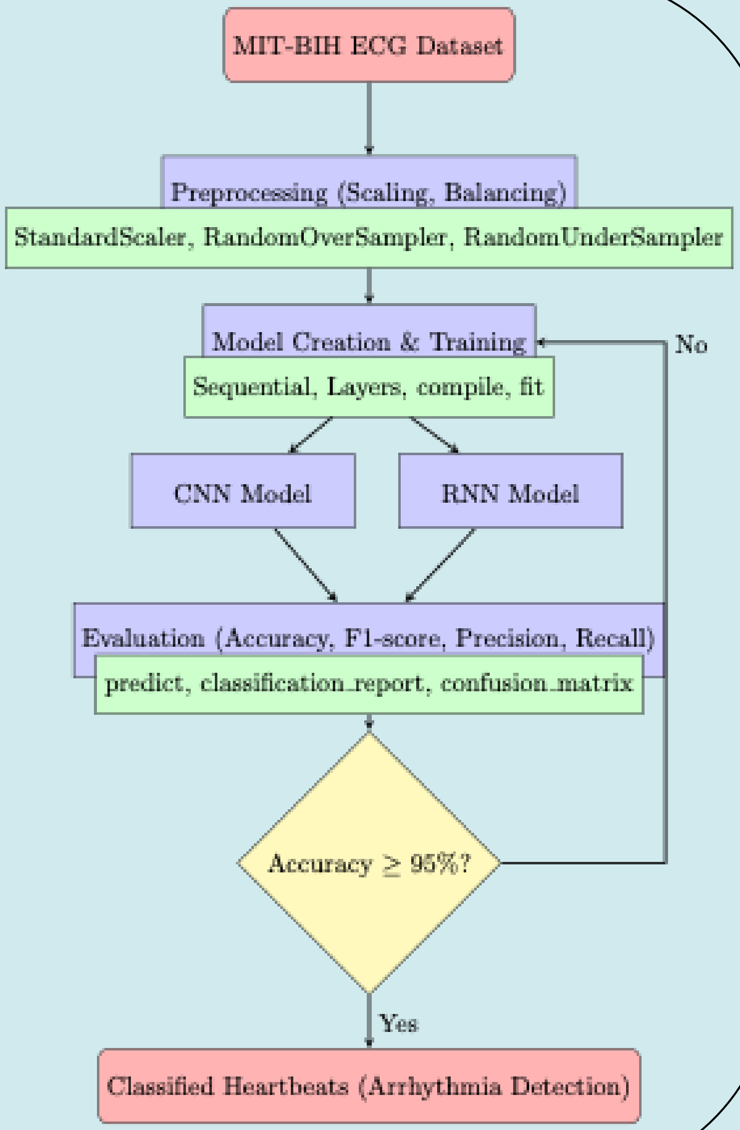
- Model Creation & Training:
  - CNN Model:
    - Input
    - 3x Convolutional Layer & MaxPooling
    - Flatten
    - Dense & Dropout
    - Output
  - CNN-RNN Model:
    - LSTM (Long Short-Term Memory):
      - LSTM (64 Units)
    - GRU (Gated Recurrent Units):
      - GRU(64 Units)
  - RELU Activation Function
  - ADAM Optimizer
  - Categorical Crossentropy
  - Reduce Learning Rate on Plateau



System Overview

- Evaluation:
  - Training Duration
  - Predict Class Labels for Test Set
    - Accuracy Score
    - Precision Score
    - Recall Score
    - F1 Score
    - Classification Report
    - Confusion Matrix

- Required Accuracy greater or equal to 95%
- Repeat, if necessary, otherwise satisfied



Algorithm Design (Iteration 1)

- Iterative Process for Model Refinement
- Based on Academic Literature and Course Concepts

Algorithm Design (Iteration 1)

- Model Comparison: Accuracy:
  - CNN: 8 min 4.4 sec 98.36%
  - LSTM: 94 min 4.9 sec 98.20%
  - GRU 74 min 33.7 sec 98.36%

- Confusion Matrices in %

CNN	N	S	V	F	U
N	99.12	0.45	0.35	0.09	0.09
S	17.09	81.47	1.26	0.00	0.18
V	2.49	0.07	96.13	1.24	0.07
F	8.02	0.00	7.41	84.57	0.00
U	0.75	0.06	0.06	0.00	99.13

LSTM	N	S	V	F	U
N	98.87	0.75	0.18	0.07	0.13
S	13.31	85.25	0.72	0.54	0.18
V	2.28	0.35	95.51	1.59	0.28
F	9.26	0.00	7.41	83.33	0.00
U	0.81	0.12	0.06	0.00	99.00

GRU	N	S	V	F	U
N	99.07	0.45	0.22	0.10	0.16
S	13.85	83.81	1.98	0.00	0.36
V	1.93	0.41	95.93	1.31	0.41
F	9.26	0.62	10.49	79.63	0.00
U	0.44	0.00	0.12	0.00	99.44

Algorithm Design (Iteration 2)

- Incorporated Residual Blocks (Main Path and Shortcut using Skip Connection)
- Model Comparison: Accuracy: Improvement
  - CNN: 120 min 41.4 sec 98.15% No
  - LSTM: 259 min 55.6 sec 98.57% Yes
  - GRU 113 min 9.3 sec 97.88% No

CNN	N	S	V	F	U
N	98.80	0.55	0.41	0.12	0.11
S	14.75	83.09	1.98	0.18	0.00
V	1.66	0.07	96.34	1.66	0.28
F	5.56	1.23	8.64	84.57	0.00
U	0.81	0.06	0.12	0.06	98.94

LSTM	N	S	V	F	U
N	99.35	0.28	0.20	0.09	0.08
S	16.01	81.83	1.62	0.36	0.18
V	2.07	0.28	96.20	1.38	0.07
F	7.41	0.62	7.41	84.57	0.00
U	0.87	0.00	0.06	0.00	99.07

GRU	N	S	V	F	U
N	98.69	0.63	0.46	0.09	0.13
S	18.53	79.86	1.26	0.36	0.00
V	1.59	0.69	95.86	1.38	0.48
F	8.02	0.62	10.49	80.86	0.00
U	0.81	0.19	0.44	0.00	98.57

Algorithm Design (Iteration 3)

- Incorporated Batch Normalization,
- Extend RNNs by introducing bidirectional layers
- Global Average Pooling instead of Flattening
- Model Comparison: Accuracy: Improved:
  - CNN: 34 min 33.9 sec 97.88% No
  - LSTM: 28 min 20.7 sec 98.20% No
  - GRU 30 min 50.9 sec 98.04% No
- Confusion Matrices in %

CNN	N	S	V	F	U
N	98.60	0.81	0.38	0.06	0.15
S	15.65	82.55	1.44	0.36	0.00
V	1.80	0.41	95.99	1.31	0.48
F	9.88	0.00	12.35	77.78	0.00
U	0.81	0.31	1.87	0.00	98.69

LSTM	N	S	V	F	U
N	98.88	0.64	0.24	0.11	0.13
S	15.29	82.37	1.80	0.54	0.00
V	2.28	0.21	96.13	1.24	0.14
F	9.26	0.00	8.02	82.72	0.00
U	0.37	0.62	0.12	0.62	99.38

GRU	N	S	V	F	U
N	98.65	0.67	0.36	0.19	0.13
S	13.67	83.81	1.80	0.36	0.36
V	1.86	0.41	95.86	1.59	0.28
F	3.70	0.62	8.64	87.04	0.00
U	0.56	0.00	0.12	0.19	99.13

Future Work & Conclusion

- Iteration 1 Models have already been promising classifiers.
- The Requirements have been met generally, a working classifier has been trained using multiple approaches.
- Nevertheless per-class performance needs improvement.
- The most efficient classifier was the CNN in Iteration 1, and the overall best classifier the CNN-LSTM in Iteration 2 using Residual Connections.
- Suggestion:
  - Perform synthetic over sampling techniques such as SMOTE or ADASYN for better class balancing
  - Focus on per-class precision and recall to meet EU AI Regulations on healthcare requirements

Creating change together.

UNIVERSITY WEST

Thank You!

Contact: lewis.richter@student.hv.se

University West, Trollhättan, Sweden

GitHub: ECGClassification.git

UNIVERSITY WEST

Creating change together.