Practical Work 1 - Heartbeat Classification: Comparison between Random Forest and 1D-CNN

I. Introduction

Electrocardiogram (ECG) signals provide a reliable means of assessing the functionality of the cardiovascular system. In this paper, I will implement and compare two approaches on the dataset downloading from kaggle - the ECG Heartbeat Categorization Dataset:
1) Machine Learning approach using RandomForest optimized with GridSearchCV
2) Deep Learning approach using a 1D-CNN combined with SMOTE to specifically address the class imbalance problem

II. Dataset Overview

The dataset consists of two parts:

- Training set: 87,554 samples with 187 features and 1 label column
- Testing set: 21,892 samples

The heartbeat categories are:

1) Normal
2) Supraventricular
3) Ventricular
4) Fusion
5) Unknown

Fig. 1: Class Distribution in Training Set

III. Class Distribution

The training set exhibits significant class imbalance:

- Normal: 72,471 samples
- Supraventricular: 2,223 samples
- Ventricular: 5,788 samples
- Fusion: 641 samples
- Unknown: 6,431 samples

IV. Methodology

A. Approach 1: Random Forest with GridSearchCV

A RandomForestClassifier was trained using GridSearchCV to optimize hyperparameters. To address class imbalance during training, the scoring metric f1_weighted was applied. The best parameters found were:

- n_estimators: 100
- max_depth: None
- min_samples_split: 5

B. Approach 2: 1D-CNN with SMOTE

To improve the recall of minority classes (Supraventricular and Fusion), I applied SMOTE to balance the training dataset. Subsequently, a 1D-Convolutional Neural Network (CNN) was designed to extract temporal features from the ECG signals efficiently.

V. Results

A. Random Forest Method Performance

The Random Forest model achieved a test accuracy of **97.42%**. However, as shown in table below, the model struggled with minority classes - likely due to the problem of class imbalance

TABLE I: Random Forest Classification Report

Class	Precision	Recall	F1-score	Support
Normal	0.97	1.00	0.99	18118
Supraventricular	0.95	**0.60**	0.74	556
Ventricular	0.98	0.89	0.93	1448
Fusion	0.82	**0.61**	0.70	162
Unknown	1.00	0.95	0.97	1608
Accuracy			**0.97**	21892
Weighted Avg	0.97	0.97	0.97	21892

B. Confusion Matrix (Random Forest)

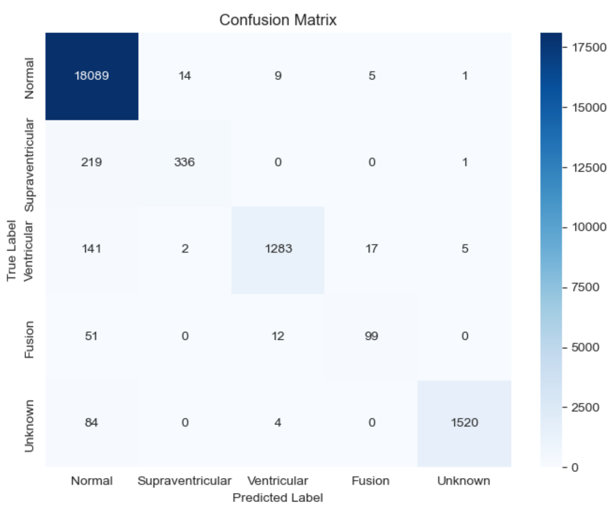

Fig. 2: Confusion Matrix of Random Fores Model

C. CNN Method Performance

The CNN model with SMOTE achieved a higher test accuracy of **98.36%**. More importantly, the Recall for minority classes improved significantly:

TABLE II: CNN Classification Report

Class	Precision	Recall	F1-score	Support
Normal	0.99	0.99	0.99	18118
Supraventricular	0.84	**0.82**	0.83	556
Ventricular	0.98	0.94	0.96	1448
Fusion	0.72	**0.83**	0.77	162
Unknown	0.99	0.99	0.99	1608
Accuracy			**0.98**	21892
Weighted Avg	0.98	0.98	0.98	21892

D. Confusion Matrix (CNN)

Fig. 3: Confusion Matrix of CNN Model (with SMOTE)

VI. Conclusion

While Random Forest with GridSearchCV provided strong overall accuracy (97.42%), it showed limitations in detecting rare classes (Recall $\approx$ 0.60). By applying SMOTE and using a Deep Learning architecture (1D-CNN), I improved the overall accuracy to **98.36%** and significantly boosted the recall for 'Supraventricular' (0.60 $\rightarrow$ 0.82) and 'Fusion' (0.61 $\rightarrow$ 0.83) categories. This demonstrates that applying oversampling methods is crucial for datasets having imbalance problem.