ECG Heartbeat Categorization

BI12-418 Doan Huu Thanh

March 20, 2024

Contents

1

Abstract

This dataset includes two sub-collections developed from the renowned datasets in heartbeat classification: the MIT-BIH Arrhythmia Dataset and the PTB Diagnostic ECG Database. Both of them contain an adequate number of samples for training deep neural networks.

The use of this dataset includes the exploration of heartbeat classification using deep neural networks. The signals correspond to electrocardiogram shapes of heartbeats and have been preprocessed and segmented, with each segment corresponding to a heartbeat.

2

Dataset

2.1 The components

This dataset consists of CSV files, each of them featuring a matrix with each row representing an example in that portion of the dataset. The last element of each row indicates which class that example belongs to. In addition, all of the samples have been cropped, downsampled and padded with zeroes where necessary to a fixed dimension of 188.

2.1.1 The Arrhythmia dataset

- Number of samples: 109446
- Number of categories: 5. Classnames: ['N': 0, 'S': 1, 'V': 2, 'F': 3, 'Q':, 4]
- Sampling frequency: 125Hz
- Data source: Physionet's MIT-BIH Arrhythimia Dataset

This sub-collection includes 2 CSV files: 'mitbih_train.csv' as the training set and 'mitbih_test.csv' as the testing set.

The training set has 87552 samples, while the testing set has 21890 samples. There are no missing values and all of the samples have been preprocessed to ensure similar dimensions.

2.1.2 The PTB Diagnostic ECG database

- Number of samples: 14552
- Number of categories: 2
- Sampling frequency: 125Hz
- Data source: Physionet's PTB Diagnostic Database

This sub-collection includes 2 CSV files: 'ptbdb_abnormal.csv' and 'ptbdb_normal.csv'.

The first file contains 10500 samples, while the latter contains 4045 samples. There are no missing values and all of the samples have been preprocessed to ensure similar dimensions.

3

Model implementation

Only 'mitbih_train.csv' and 'mitbih_test.csv' are utilised for this implementation.

3.1 Data preparation

Since the last value of each row indicates the class to which each sample belongs, we sliced the last column of each dataset to create the target labels. The remaining data is divided into a training set and a validation set with a ratio of 80/20.

By visualizing the samples, we observe that both the training and testing sets are imbalanced, with 82.8% of the samples being Normal beats (the 'N' class) while the remaining four classes only account for 17.2%.

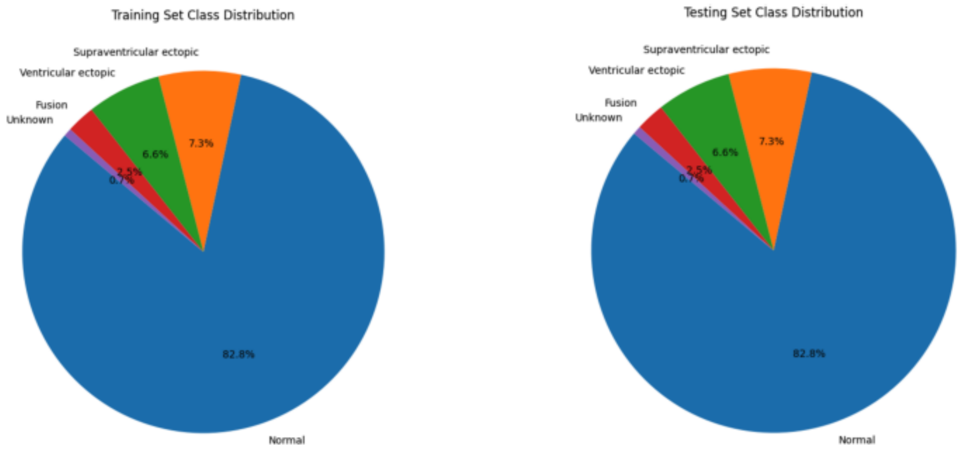

Figure 3.1: Data distribution

3.2 Model architecture

We created a simple neural network using the Sequential class from the Keras library. The model contains two stages and a final fully connected layer to classify the beat types.

Since the heartbeat data are 1D signals, we utilize Conv1D layers. The first block contains a Conv1D layer with 64 filters, kernels of size 5x1 and ReLU activation, followed by a MaxPooling1D layer with a pool size of 2. At the end of the block we use a dropout rate of 0.1.

The second block contains another Conv1D layer with 128 filters, kernels of size 5x1 and ReLU activation, followed by a MaxPooling1D layer with a pool size of 2. The data will then be flatten before being fit into a fully connected dense layer with 128 neurons. Here we use a dropout rate of 0.2.

Finally the final fully connected dense layer with 5 neurons representing 5 classes will be used to classify the heartbeat signals.

3.3 Model training

Since we only have a training set and a testing set at the beginning, we further split the training set into a smaller training set and an additional validation set with a ratio of 80/20. After this, we are fitting both sets into the model to commence training.

We set the model to be trained for 20 epochs with a batch size of 64. We also used ReduceLROnPlateau to monitor the learning rate.

3.4 Evaluation and results

After the training process, we obtained the results as shown in the plots below:

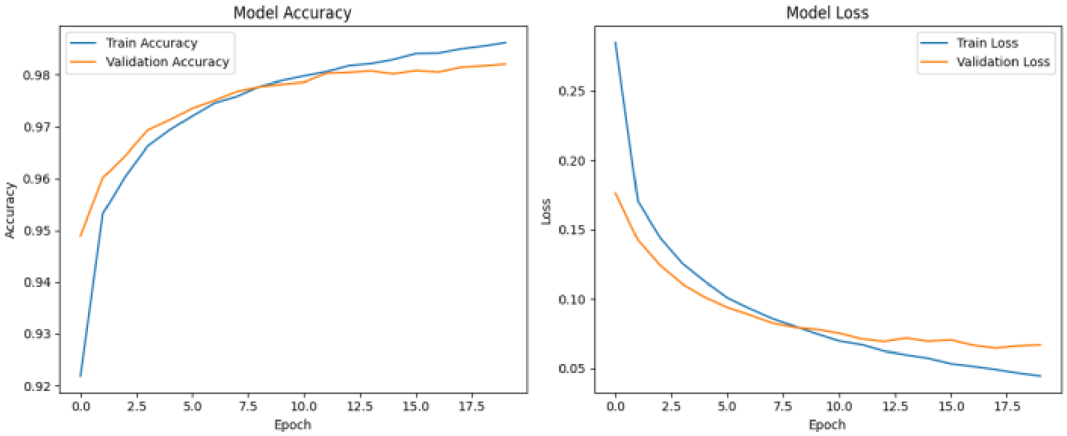

Figure 3.2: Training results

Both the training and validation accuracy increased significantly within the first 5 epochs before gradually improving during the remaining 15 epochs. A similar trend can be seen with both the training and validation loss.

At the end of the training process, it is implied from both of the plots that the difference between the training and validation accuracy is relatively small, less than 0.01% and the same applies for the loss values. This indicates a positive results which shows a low risk of overfitting.

The performance of the model on the seperated test set is shown below:

Accuracy	Precision	Recall	F1-score
0.98	0.93	0.85	0.88

The confusion matrix of the model:

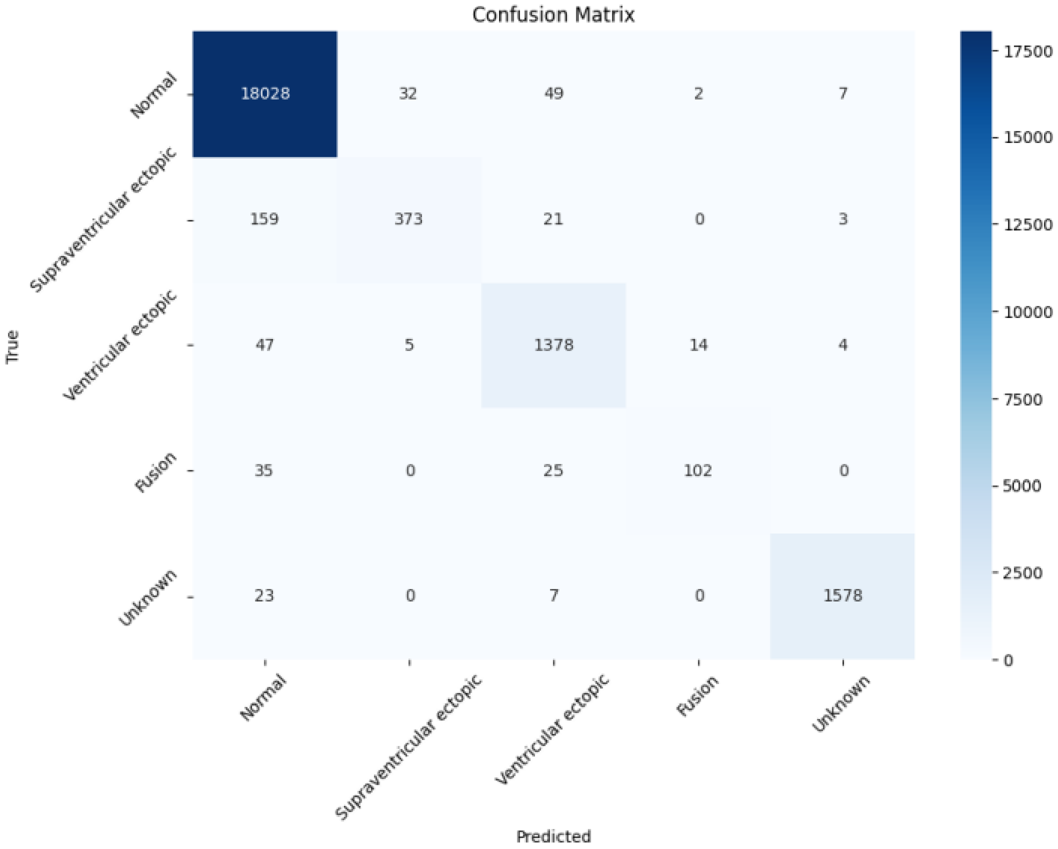

Figure 3.3: Confusion matrix of the custom model

From the matrix, we can see that the Normal beats are correctly classified the most since this class accounts for around 80% of the data. However, there are also a notable amount of incorrect classifications that are confused with the remaining 4 classes. Also for the remaining 4 classes, the model perform relatively well, as shown by the small number of confused classifications with other classes. In general, the model demonstrated an impressive performance and generalized well on the dataset.