ECG SIGNLA PREDICTION

1/28/2025
Deep Learning
DATASETS: 1&2

Rudmehr Aghakhani
ID:4003663002

Dr. Kiani

Model Overview

In this project, we built and optimized a multi-class classification model for time-series data, particularly focusing on ECG signal data. Below is a detailed overview of the steps and techniques we implemented to enhance the model's performance.

1. Data Preprocessing

The data consisted of two CSV files: one for training and the other for testing. Each dataset was headless, with the last column representing the class labels (five classes in total). The initial steps of preprocessing involved:

- **Data Loading:** We loaded the datasets using Pandas and merged them for initial exploration.
- **Data Splitting:** The training set was further split into 80% for training and 20% for validation to ensure the model generalized well to unseen data.
- **Fourier Transform:** ECG signals were processed using the Fast Fourier Transform (FFT) to extract frequency-domain features, which are often more informative for time-series classification tasks.
- **One-Hot Encoding:** Labels were converted to categorical format using one-hot encoding for multi-class classification.

2. Addressing Class Imbalance

Class imbalance is a common problem in multi-class datasets and can lead to biased models. To handle this issue, we applied **Random Oversampling** using the `RandomOverSampler` class from the `imblearn` library.

- This technique duplicated samples from minority classes to balance the class distribution.
- Oversampling was performed before one-hot encoding to ensure consistency in label processing.

3. Model Architecture

We designed a deep learning model incorporating both convolutional and attention-based layers to capture spatial and temporal dependencies within the data.

Conv1D Layers

- We used multiple `Conv1D` layers with ReLU activation and batch normalization.
- Max pooling layers were applied to reduce the sequence length and computational complexity.
- These convolutional layers were critical for capturing local patterns in the ECG signals.

Positional Encoding and Attention Mechanism

- A custom **Positional Encoding Layer** was added to provide positional information to the model since attention mechanisms do not inherently capture sequence order.
- **Multi-Head Attention Layers** were implemented to enable the model to focus on different parts of the sequence simultaneously.
- A custom **Mask Adjustment Layer** ensured that padding values did not interfere with attention calculations, improving the model's efficiency.

Transformer Encoder Stack

- We incorporated multiple transformer encoder layers with residual connections and feed-forward networks to capture complex dependencies in the data.

Regularization and Dropout

- To mitigate overfitting, we applied **L2 regularization** with a value of 0.01-.0.02, which penalizes large weights during training.
- Dropout layers with a 40% rate were introduced to further reduce overfitting by randomly dropping nodes during each training step.

Classification Layers

- We added fully connected dense layers, culminating in an output layer with five units (one for each class) and a softmax activation function for multi-class classification. And for two class problem we used sigmoid.

4. Training Optimization

To ensure effective training, we used the following optimization techniques:

- **Learning Rate Adjustment:** We added a learning rate scheduler callback to reduce the learning rate dynamically when the validation loss plateaued.
- **Gradient Clipping:** We employed `clipnorm=1.0` to prevent exploding gradients, which can destabilize training.

- **Adam Optimizer:** The model was compiled with the Adam optimizer, which is known for its adaptive learning rate capabilities.

5. Training & Prediction results:

Two Classes

```
Epoch 37/45
209/211 ━━━━━━━━━━━━━━━━━━━━ 0s 8ms/step - accuracy: 0.9995 - loss: 0.0184
Epoch 37: ReduceLROnPlateau reducing learning rate to 1e-06.
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 9ms/step - accuracy: 0.9995 - loss: 0.0184 - val_accuracy: 0.9725 - val_loss: 0.1466 - learning_rate: 1.9531e-06
Epoch 38/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 10ms/step - accuracy: 0.9992 - loss: 0.0191 - val_accuracy: 0.9725 - val_loss: 0.1474 - learning_rate: 1.0000e-06
Epoch 39/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 10ms/step - accuracy: 0.9992 - loss: 0.0204 - val_accuracy: 0.9729 - val_loss: 0.1476 - learning_rate: 1.0000e-06
Epoch 40/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 9ms/step - accuracy: 0.9997 - loss: 0.0175 - val_accuracy: 0.9729 - val_loss: 0.1474 - learning_rate: 1.0000e-06
Epoch 41/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 8ms/step - accuracy: 0.9994 - loss: 0.0174 - val_accuracy: 0.9734 - val_loss: 0.1481 - learning_rate: 1.0000e-06
Epoch 42/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 3s 8ms/step - accuracy: 0.9994 - loss: 0.0194 - val_accuracy: 0.9734 - val_loss: 0.1477 - learning_rate: 1.0000e-06
Epoch 43/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 9ms/step - accuracy: 0.9996 - loss: 0.0181 - val_accuracy: 0.9729 - val_loss: 0.1486 - learning_rate: 1.0000e-06
Epoch 44/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 3s 10ms/step - accuracy: 0.9994 - loss: 0.0186 - val_accuracy: 0.9734 - val_loss: 0.1486 - learning_rate: 1.0000e-06
Epoch 45/45
211/211 ━━━━━━━━━━━━━━━━━━━━ 2s 11ms/step - accuracy: 0.9996 - loss: 0.0181 - val_accuracy: 0.9729 - val_loss: 0.1487 - learning_rate: 1.0000e-06
91/91 ━━━━━━━━━━━━━━━━━━━━ 3s 15ms/step

Classification Report:
              precision    recall  f1-score   support

         0.0       0.95      0.95      0.95       809
         1.0       0.98      0.98      0.98      2102

    accuracy                           0.97      2911
   macro avg       0.97      0.97      0.97      2911
weighted avg       0.97      0.97      0.97      2911
```

Five Classes

```
4530/4530 ━━━━━━━━━━━━━━━━━━━━ 83s 12ms/step - accuracy: 0.9990 - loss: 0.0079 - val_accuracy: 0.9762 - val_loss: 0.1870 - learning_rate: 6.2500e-05
Epoch 27/30
4530/4530 ━━━━━━━━━━━━━━━━━━━━ 82s 12ms/step - accuracy: 0.9993 - loss: 0.0064 - val_accuracy: 0.9766 - val_loss: 0.1897 - learning_rate: 3.1250e-05
Epoch 28/30
4530/4530 ━━━━━━━━━━━━━━━━━━━━ 57s 13ms/step - accuracy: 0.9994 - loss: 0.0065 - val_accuracy: 0.9764 - val_loss: 0.1930 - learning_rate: 3.1250e-05
Epoch 29/30
4529/4530 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step - accuracy: 0.9995 - loss: 0.0062
Epoch 29: ReduceLROnPlateau reducing learning rate to 1.5625000742147677e-05.
4530/4530 ━━━━━━━━━━━━━━━━━━━━ 57s 13ms/step - accuracy: 0.9995 - loss: 0.0062 - val_accuracy: 0.9774 - val_loss: 0.1965 - learning_rate: 3.1250e-05
Epoch 30/30
4530/4530 ━━━━━━━━━━━━━━━━━━━━ 82s 13ms/step - accuracy: 0.9995 - loss: 0.0059 - val_accuracy: 0.9770 - val_loss: 0.1914 - learning_rate: 1.5625e-05
685/685 ━━━━━━━━━━━━━━━━━━━━ 6s 5ms/step
Confusion Matrix:
[[17903    86    81    14    34]
 [  121   421    13     0     1]
 [   82     2  1346    14     4]
 [   25     0    12   125     0]
 [   36     0    11     0  1561]]

Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.99      0.99     18118
           1       0.83      0.76      0.79       556
           2       0.92      0.93      0.92      1448
           3       0.82      0.77      0.79       162
           4       0.98      0.97      0.97      1608

    accuracy                           0.98     21892
   macro avg       0.91      0.88      0.89     21892
weighted avg       0.98      0.98      0.98     21892
```

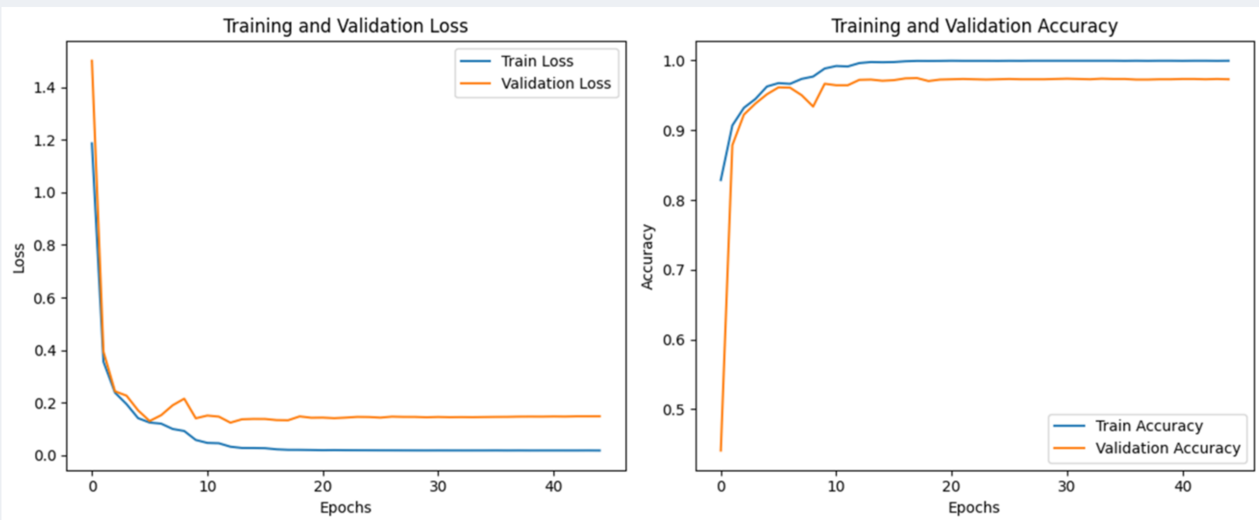
6. Model Evaluation and Performance Analysis

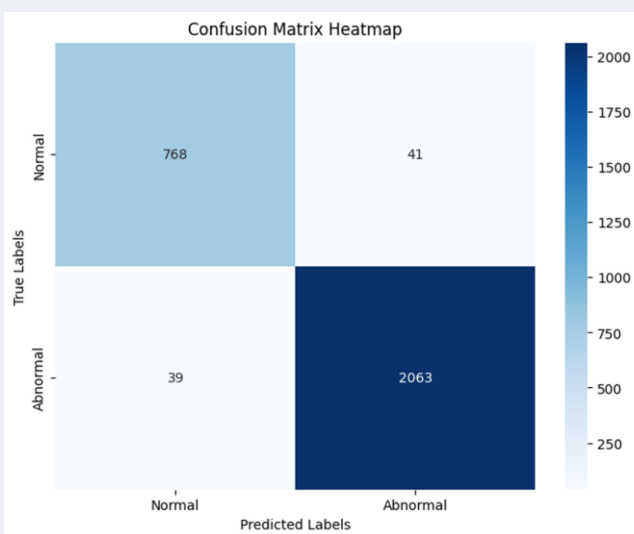
- The model was trained for 30-45 epochs with a batch size of 64.
- Training and validation accuracy and loss were monitored.
- The validation loss stabilized at approximately 0.13, while the training loss was around 0.02 for two classes dataset and for the second one; While for the five classes dataset its validation stabilized at 0.19 and its trainng became 0.005 (A little overfitting is observable).

Confusion Matrix and Classification Report

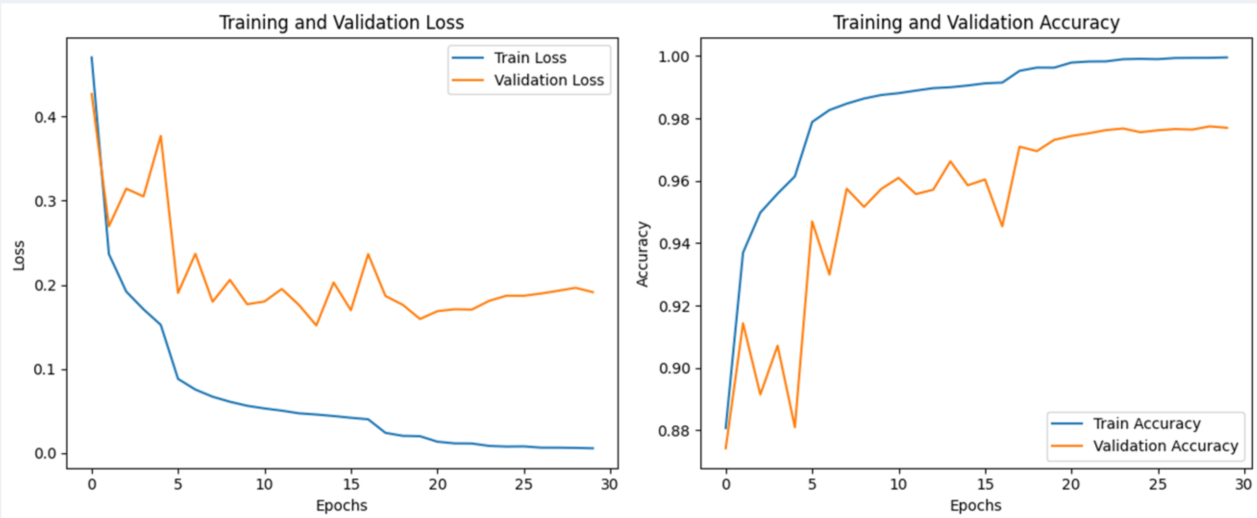
- We evaluated the model on the test set and generated a confusion matrix to analyze class-specific performance.
- A classification report was produced, showing precision, recall, and F1-score for each class.

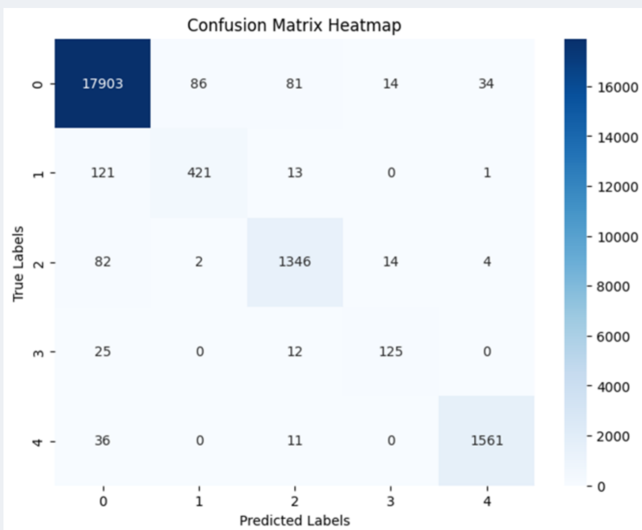
- Here are the results for the two classes:





- And for five classes:





7. Key Observations and Improvements

- The difference between training and validation loss indicated some overfitting, even with dropout and regularization.
- The addition of attention mechanisms and positional encoding significantly improved the model’s ability to handle sequential dependencies.
- Oversampling effectively balanced the class distribution, leading to better performance on minority classes.
- Further fine-tuning of hyper parameters, including L2 regularization and dropout rates, could improve generalization.

Conclusion

Overall, the model successfully leveraged both convolutional and attention-based layers to classify ECG signals into five classes. The combination of effective preprocessing, architecture design, and training optimizations contributed to its performance. Further experimentation and refinement can yield even better results.

Thank You
Rudmehr Aghakhani

THE END