Deep Multi-label Multi-instance Classification on 12-Lead ECG

Yingjing Feng[1,2], Edward Vigmond[1,2]

[1] IHU Liryc, Electrophysiology and Heart Modeling Institute, fondation Bordeaux Université, Pessac-Bordeaux, France
[2] Univ Bordeaux, IMB, UMR 5251, Talence, France

Abstract

As part of the PhysioNet/Computing in Cardiology Challenge 2020, we developed an end-to-end deep neural network model, MIC-ResNet, requiring minimal signal pre-processing, for identifying 27 cardiac abnormalities from 12-lead ECG data. Our team, ECGLearner, received a score of 0.539 ± 0.114 *using 5-fold cross-validation on the full training data, and a score of* 0.486 *on the full test data, and we ranked 35th out of the 100 teams who entered the official stage that participated in this year's Challenge.*

1. Introduction

Cardiovascular diseases are the primary cause of death, and they greatly impact daily life across all demographics. The ECG signal is a common and important screening and diagnostic tool for heart conditions. Different cardiovascular diseases have different mechanisms, resulting in different ECG morphologies. Deep neural networks can learn features of the different conditions directly from ECGs, a large data set, and will hopefully achieve cardiologist-level ECG recognition [1].

The PhysioNet/Computing in Cardiology Challenge 2020 focused on automated, open-source approaches for classifying multiple cardiac abnormalities from 12-lead ECG [2, 3]. In the challenge, we applied a novel deep learning model called MIC-ResNet, which combines ResNet [4] for time series and multi-instance classification (MIC) to classify multi-center patient ECG for 27 different conditions.

2. Methods

As shown in Fig. 1, our MIC-ResNet comprises three major components: an encoder module based on 1D ResNet; a multi-instance classification (MIC) module; and a decoder module to produce an output of 27 classes going through a sigmoid function.

The only preprocessing step that we performed was to filter the ECG by applying a fourth-order Butterworth filter with a pass band of 0.5 to 50Hz for each lead of each patient signal. We did not normalize the signal, as we believed that preserving amplitude of raw ECG signal was important for some conditions such as low QRS voltages.

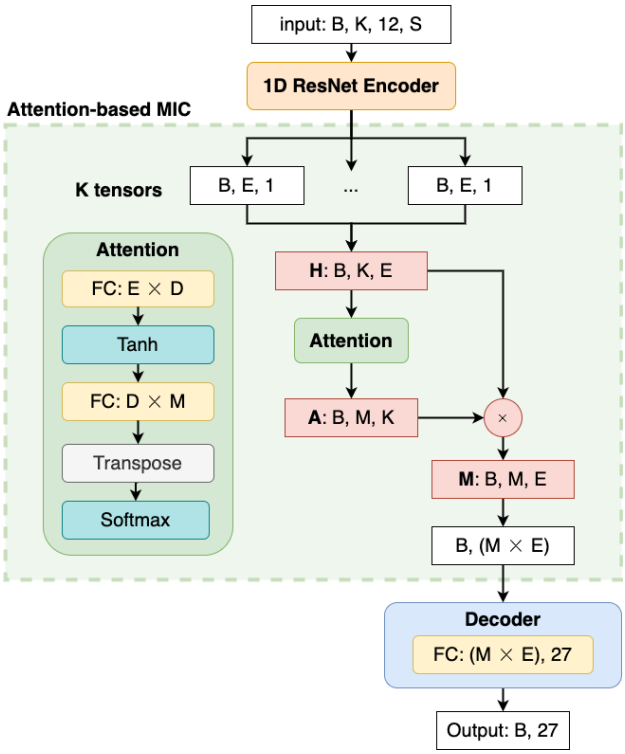

Figure 1: Architecture of MIC-ResNet

2.1. 1D ResNet Encoder

ResNet [4] is the state-of-the-art deep network for multiple types of data, from images to time series [5, 6]. There are also a multiple of works that use 1D ResNet in classifying ECG [1]. It benefits from a *shortcut* module, which enables the network to go deep, whilst remaining relatively low in complexity, thereby making the learning easier.

We used a customized 1D ResNet as an encoder to trans-

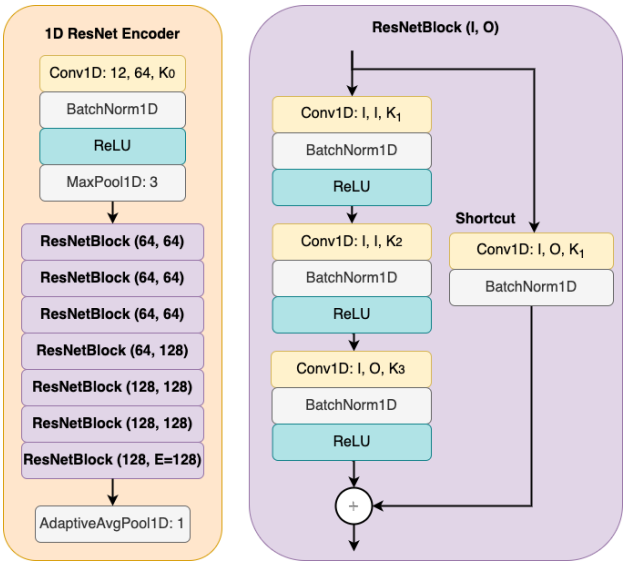

Figure 2: Encoder module in the MIC-ResNet

form a 12-channel ECG segment of S samples, to a lower-dimensional (E-dimensional) embedding, adapted from the original 2D ResNet [4]. The encoder was composed of a 1D convolutional layer (*Conv1D*) which took I, O as the input and output channel sizes, and K_0 as the kernel size, batch normalization (BatchNorm1D) [7], a non-linear activation function (*ReLU*) [8], a max pooling function (*MaxPool1D*) with a kernel size of P_0, and then followed by three building blocks (*ResNetBlocks*), where each block took an input signal with I input channels and produced O output channels. The kernel sizes $K_1, K_2,$ andK_3 for the three *Conv1D* layers in the *ResNetBlocks* came from a strong baseline 1D ResNet model for time-series classification [6]. An adaptive average pooling (*AdaptiveAvgPool1D*) with output length of 1 was placed at the end, to automatically select the stride and the kernel size in order to produce outputs of E length-one channels.

2.2. Attention-based MIC

MIC refers to a type of classification problems in which the data samples are instances in bags, and a label is only available for each bag rather than for each instance. For conditions that do not manifest themselves in a rarer event, such as premature atrial contraction, it can be formulated that the whole ECG recording is made of several instances, and PAC only occurs in a subset of them. For binary classification, the MIC pooling means the probability (or label) of the bag is the maximum of the instance probability (or label). By using a bag of K segments of S samples to represent a ECG recording of various lengths, we have spanned our search range of the ECG from K samples to $K \times S$ samples with the same encoder, without assuming that each fixed-length segment was positive.

We adopted an attention-based MIC framework proposed in [9]. For a bag of K instances going through the encoder, we obtained K embeddings as $H = \{h_1, h_2, \ldots, h_K\}$. The MIC pooling was then

$$z = \sum_{k=1}^{K} a_k h_k \tag{1}$$

where

$$a_k = \frac{\exp\{\mathbf{w}^T \tanh \mathbf{V} h_k^T\}}{\sum_{j=1}^{K} \exp\{\mathbf{w}^T \tanh \mathbf{V} h_k^T\}} \tag{2}$$

The attention module was made of two fully connected (*FC*) layers with a $\tanh(\cdot)$ layer in the middle, where the first *FC* layer was to learn the weight $\mathbf{V} \in \mathcal{R}^{D \times E}$, and the second to learn the weight $\mathbf{w} \in \mathcal{R}^{D \times M}$, together with the transpose operation and the *Softmax* layer, implemented Eq (2), and K was another hyperparameter to be optimized. A tensor multiplier $M = A \times H$ implemented Eq (1), and the resulting M went through a *FC* decoder to produce an output of C dimensions.

2.3. Implementation

As all patient ECGs were sampled at 500Hz, each containing a varying number of at least 2500 samples, we picked $S = 3000$ samples representing 6 second intervals as the training input to the ResNet 1D, so the bag input had dimensions of $(B, K, 12, S)$. Zeros were padded on the end of recordings with less than S samples. To augment the training set, we randomly sampled K instances of S samples for a training input across the whole ECG recording, whereas a validation input was composed of evenly sampled K instances of S samples.

We represented the label of each sample as $\mathbf{y} = [y_1, y_2, \ldots, y_C]$, where $C = 27$ is the total number of scored classes, and $y_i = 1$ if class i is positive and 0 otherwise. For those classes with an equivalent class, we relabelled them as positive in an entry if their equivalent class was positive. A multi-label stratified 5-fold cross-validation [10] (iterative-stratification Python package version 0.1.6) was applied on each of the six training datasets in Table 2 of [3] to constitute the full training-validation set, so that the training and the validation sets in each fold have similar class distribution. This distribution is similar across different folds, keeping performance stable between different folds.

A binary cross entropy loss (*BCELoss*) was used as the optimization target for the multi-label classification. The total *BCELoss* was defined as the average of sample *BCELoss*, and for each sample of the network output

Hyperparameters	Value
Segment length (S)	3000
Number of segments in a bag (B)	5
Positive class weight (p)	2
Encoder first Conv1D and MaxPool1D (K_0)	7
Encoder first MaxPool1D kernel (P_0)	3
ResNetBlock kernels (K_1, K_2, K_3)	7, 5, 3
ResNetBlock input output channels	see Fig.2
Parameter for attention (D, M)	64, 32
Smoothing term (γ)	1

Table 1: List of hyperparameters.

$\mathbf{x} = [x_1, x_2, \ldots, x_C]$, the *BCELoss* of each sample was:

$$l = -\sum_{i=1}^{C} w_i \left[py_i \cdot \log \sigma(x_i) + (1 - y_i) \cdot \log(1 - \sigma(x_i)) \right].$$

To consider for class imbalance, the class weight for each class i was defined as in [11],

$$w_i = \log \frac{N - n_i + \gamma}{n_i + \gamma}, N = \sum_{j=1}^{C} n_j,$$

where i is the number of positive instances of class i, and γ is a smoothing term, a larger class weight was given to classes with small samples, and got optimized at a higher priority. A positive weight $p = 2$ was added to the *BCELoss* for all classes to give a higher weight for recall than precision.

We used PyTorch version 1.4, CUDA version 10.2. We used the Adam optimizer [12] with a learning rate of 0.01, and rescaled with a factor of 0.1 when the validation loss reached a plateau for 10 epochs. The training stopped when there was no reduction in the validation loss for over 20 epochs. We used mini-batch gradient descent with a batch size of 64. We trained on a Quadro RTX 8000 Graphic Processing Unit, and each fold stopped at around 55 epochs and three hours. After five folds were trained, we averaged the validation loss, and computed the optimal epoch producing the lowest averaged validation loss. We then trained on the whole dataset for the optimal epoch. All hyperparameters in our algorithm are summarized in Table 1.

3. Results

We compared our results with using instance-wise 1D ResNet composed of only the encoder and decoder in Fig. 1. During training, one segment of S samples was drawn randomly from ECG for each training entry and one central segment of S samples for each validation entry during training. During validation, we used the same K instances as in MIC and made predictions for the ECG on two mode: the *First* mode used the output of first instance, and the *Max* mode used the maximal amongst the K instances. The competition metrics ($\beta = 2$) with the geometric mean of $geometry = \sqrt{F_\beta G_\beta}$ are shown in Table. 2.

We calculated a multi-label confusion matrix, where its diagonal holds the true positive rate, and the rest shows the false positive rate. The per-class performance and the confusion matrix of the MIC model are shown in Fig. 3 and Fig. 4.

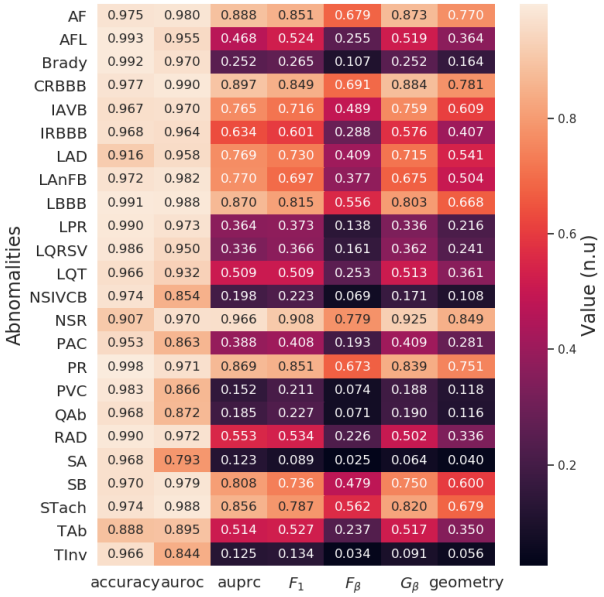

Figure 3: Per-class metric over five folds

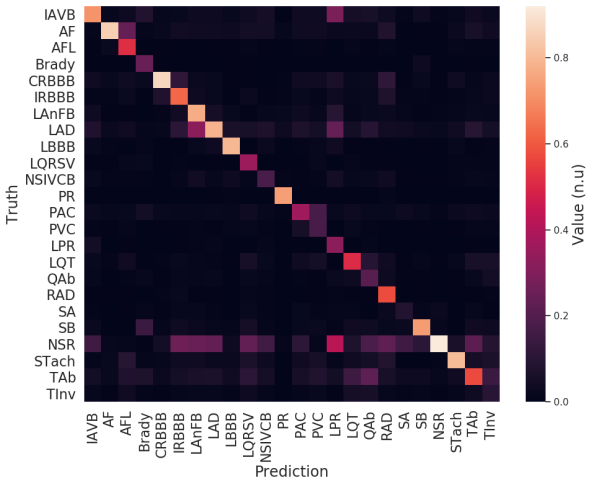

Figure 4: Multi-label confusion matrix over five folds.

	accuracy	auroc	auprc	F_1	F_β	G_β	challenge metric
MIC	$\mathbf{0.936 \pm 0.002}$	$\mathbf{0.551 \pm 0.003}$	0.523 ± 0.010	$\mathbf{0.538 \pm 0.004}$	0.530 ± 0.007	$\mathbf{0.325 \pm 0.004}$	0.539 ± 0.014
First	0.932 ± 0.002	0.544 ± 0.007	$\mathbf{0.528 \pm 0.006}$	0.519 ± 0.001	0.506 ± 0.006	0.316 ± 0.008	0.517 ± 0.004
Max	0.935 ± 0.002	0.550 ± 0.006	0.494 ± 0.012	0.529 ± 0.005	$\mathbf{0.534 \pm 0.006}$	$\mathbf{0.325 \pm 0.004}$	$\mathbf{0.556 \pm 0.014}$

Table 2: Challenge metrics over five folds

4. Discussion and Conclusions

We developed a multi-label classifier for 12-lead ECGs with an attention-based MIC, which received a score of 0.534 ± 0.113 using 5-fold cross-validation on the full training data, and a score of 0.486 on the full test data.

In Table 2, although *Max* mode received the best challenge score, biasing towards recall. By aggressively selecting the maximal probability, *Max* resulted in a low precision for auprc and F_1, whereas *First* received the best auprc but the worst challenge score. On the other hand, MIC took a balance between precision and recall, and we believe this is important. In Fig. 3, the classifier achieved an accuracy near to 1 in practically all cases (except for 0.888 for TAb), and auroc ≥ 0.8. The geometry score shows that our model works the best for AF, CRBBB, NSR, and PR, which are all conditions exhibiting abnormality in each beat, whereas the worst were mainly conditions that did not occur in each beat (e.g. PVC), or had abnormal amplitudes (e.g. Tinv), durations (e.g. LPR and Brady), and abnormalities with multiple underlying causes (eg. NSIVCB and QAb). In Fig. 4, the inter-class misclassification occurred the most from NSR as the ground-truth, followed by TAb to QAb, between PAC and PVC, and a few abnormalities were mistaken as LPR and TInv.

In conclusions, we developed a deep neural network combining 1D ResNet with MIC to predict for multiple cardiac abnormalities from 12-lead ECG, which received a score of 0.539 ± 0.114 using 5-fold cross-validation on the full training data and a score of 0.486 on the full test data. Future improvements include adding more weights in complex abnormalities, optimising the hyperparameters in Table 1 with cross-validation, and further data augmentation with added noise.

Acknowledgments

Funding has been received from the European Union Horizon 2020 research and Innovation programme "Personalised In-silico Cardiology (PIC)" under the Marie Sklodowska-Curie grant agreement No 764738, and the French National Research Agency (ANR-10-IAHU-04).

References

[1] Hannun AY, Rajpurkar P, Haghpanahi M, Tison GH, Bourn C, Turakhia MP, Ng AY. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature Medicine ;(1):65–69. ISSN 1546-170X.

[2] Goldberger AL, Amaral LA, Glass L, Hausdorff JM, Ivanov PC, Mark RG, Mietus JE, Moody GB, Peng CK, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation 2000;101(23):e215–e220.

[3] Perez Alday, EA, Gu A, Shah A, Robichaux C, Wong A, Liu C, Liu F, Rad B, Elola A, Seyedi S, Li Q, Sharma A, Clifford G, Reyna M. Classification of 12-lead ecgs: the physionet/computing in cardiology challenge 2020. Physiological Measurement ;(Under Review) 2020.

[4] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition dec;.

[5] Wang Z, Yan W, Oates T. Time series classification from scratch with deep neural networks: A strong baseline. In Proceedings of the International Joint Conference on Neural Networks. ISBN 9781509061815; 1578–1585.

[6] Ismail Fawaz H, Forestier G, Weber J, Idoumghar L, Muller PA. Deep learning for time series classification: a review. Data Mining and Knowledge Discovery ;(4):917–963. ISSN 1573756X.

[7] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In 32nd International Conference on Machine Learning, ICML 2015, volume 1. ISBN 9781510810587, 2015; 448–456.

[8] Agarap AF. Deep Learning using Rectified Linear Units (ReLU) mar;.

[9] Ilse M, Tomczak JM, Welling M. Attention-based deep multiple instance learning. In 35th International Conference on Machine Learning, ICML 2018, volume 5. ISBN 9781510867963, 2018; 3376–3391.

[10] Sechidis K, Tsoumakas G, Vlahavas I. On the Stratification of Multi-label Data. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), number PART 3. ISBN 9783642238079; 145–158.

[11] Puntonet CG, Lang EW. Blind source separation and independent component analysis, 2006.

[12] Kingma DP, Ba JL. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings. 2015; .

Address for correspondence:

Yingjing Feng
IHU Liryc, F-33600 Pessac-Bordeaux, France
yingjing.feng@ihu-liryc.fr