Hybrid Arrhythmia Detection on Varying-Dimensional Electrocardiography: Combining Deep Neural Networks and Clinical Rules

Jingsu Kang[1], Hao Wen[2]

[1]Tianjin Medical University, Tianjin, China
[2]LMIB, School of Mathematical Sciences, Beihang University, Beijing, China

Abstract

Aim: This study aims to develop effective approaches for the detection of cardiac arrhythmias from varying-dimensional electrocardiography (ECG) for the problem raised in the PhysioNet/Computing in Cardiology Challenge 2021 (CinC2021), taking advantage of both deep neural networks (DNNs) and insights from clinical diagnostic criteria.

Methods: 26 classes (equivalent classes are counted one) of ECG arrhythmias are divided into two categories. Detectors are manually designed for 5 classes in the category with clear clinical rules. The rest classes with subtle morphological and spectral characteristics are classified by DNNs. To make the networks capable of capturing features of different scopes, we use multi-branch convolutional neural networks (CNNs) as backbone, each with different receptive fields via dilated convolutions. Considering ECGs' varying dimensionality, we design a novel structure for the networks: convolutions are grouped with group number equaling the number of leads. Outputs from DNNs and from manual detectors are merged to give final predictions.

Results: Although we (team name "Revenger") did not officially rank (the code failed to complete on the 12-lead test set), we received test scores of 0.33, 0.35, 0.33, 0.33, and 0.33 on the 2-lead, 3-lead, 4-lead and 6-lead test sets respectively.

Conclusion: The proposed hybrid method is effective for establishing auxiliary diagnosis systems, and the reduced-lead ECGs are sufficient for such systems.

1. Introduction

Heart disease is the leading cause of death worldwide [1]. Electrocardiogram (ECG), as a physiological signal that reflects the electrical activity of the myocardium, is widely adopted for screening heart diseases. Despite the rapid growth of the application of ambulatory ECG and other methods, the standard 12-lead ECG is still the most widely accepted in the clinical practice. However, the 12-lead ECG equipments for real-time monitoring of the cardiac electrical activities, especially the 24-hour Holter monitoring, usually collect an enormous amount of data. Therefore, how to properly process these ECG signals, so as to be rapid, accurate and complete for the diagnosis and early intervention of heart diseases, is still an urgent problem that needs to be solved.

For this reason, many algorithms have been proposed to accurately and automatically assist in the diagnosis of cardiac electrical abnormalities. Among them, deep neural networks (DNNs) have achieved great success [2–4] in recent years and have been dominating in this research field. These models are claimed to be able to achieve very high precision, even comparable to cardiologists [2]. However, these work have their drawbacks. For example, their models have only been validated on just a few typical ECG arrhythmias (for example atrial fibrillation, ventricular tachycardia, etc. [2]). Data redundancy is another aspect that these work did not consider [3, 4]. Hence the problem raised in the begging of the paper still needs to be studied in more depth.

In this paper, our effort of tackling this problem, which uses DNNs in combination with clinical rules providing better interpretability, will be described. Following the mission of The PhysioNet/Computing in Cardiology Challenge 2021 (CinC2021) [5–7], we apply our hybrid method on the problem of classifying a very broad range of cardiac abnormalities, instead of just several typical ones, using reduced-lead ECGs which largely reduce the data amount while at the cost of only holding partial information of the standard 12-lead ECGs. The 5 lead-sets reduced from the standard 12-lead are listed as follows.

- Twelve leads: I, II, III, aVR, aVL, aVF, V1, V2, V3, V4, V5, V6
- Six leads: I, II, III, aVR, aVL, aVF
- Four leads: I, II, III, V2
- Three leads: I, II, V2
- Two leads: I, II

We find that our method can achieve performances comparable to the standard 12-lead ECGs on the reduced-lead

ECGs, even in the extreme 2-lead case. This might suggest that the standard 12-lead ECGs might be highly redundant for the diagnosis of a very large proportion of the cardiac abnormalities. Another point is that our DNN models have acceptable performance on the detection of this wide range cardiac abnormalities, hence can serve as eligible assists for the cardiologists.

Our findings will be introduced in more details in the rest of the paper, which is organized as follows: Section 2 explains the selection of data processing methods, including methods of data preprocessing and data augmentation. Section 3 introduces the deep learning models with novel network architectures. Section 4 that follows discusses the training setups for the neural network models, including the selection of loss functions, optimizers and learning rate schedulers, etc. In order to assist the deep learning models, we design several detectors based on clinical rules for easily distinguishable abnormal ECG signals in section 5. The performance of our solution to the problem of CinC2021 is summarized in section 6. Section 7 contains our thorough ablation studies for the selection of model structures, loss functions and data augmentation methods, etc. Section 8 concludes this work.

2. Data Processing

2.1. Partitioning and Selection of Data

Coarsely, the scored ECG arrhythmias of the challenge can be divided into 2 categories. Most classes (21 classes, with equivalent classes counted as one) are characterized by subtle morphological and spectral changes. This majority of ECG arrhythmias are detected (classified) using DNNs, which are described in Section 3. The rest 5 classes ("Brady", "LAD", "RAD", "LQRSV", "PR") have clear and easy-to-describe clinical diagnostic criteria. For these ECG arrhythmias, manually designed detectors from clinical rules are included as a part of our solution. These detectors are described in more details in Section 5.

Although there are totally 132 classes of abnormalities available in the challenge database, only the scored ones are included for the development of our challenge approach. ECG records with no scored classes are discarded. We also exclude the StPetersburg subset (the INCART dataset [8]) from training the models for several reasons. Most importantly, these records are 30 minutes long with only at most 2 classes of scored abnormalities, which is too coarse. Second, each of the 9 scored classes in the StPetersburg subset constitutes less than 0.2%, by counting the number of occurrence, of the total challenge database, which is almost neglectable.

2.2. Preprocess and Data Augmentation

To make training and inference data in better consistency, data are filtered using a Butterworth filter of order 5 and passband 0.5 Hz - 60 Hz, after which baseline wander and high frequency noises are removed. The high cutoff frequency is slightly higher than usual due to the fact that the distinguishing characteristics of the pacing rhythm ("PR") are vertical spikes of very short duration.

For training DNNs, the ECGs are resampled to 500 Hz, cropped or zero-padded to ensure 10-second length (5000 sample points) to utilise mini-batch (parallel) training. ECGs with length $\geq$ 14s will be sliced into multiple training examples of overlap length 4s with the same label. Considering the existence of the class "LQRSV" (low qrs voltages) which is directly related to the magnitude of the signals (absolute values in voltages), we experimented both with and without z-score normalization of the input ECGs (denoted $\mathbf{x}$) defined as follows

$$\frac{\mathbf{x} - \mathrm{mean}(\mathbf{x})}{\mathrm{std}(\mathbf{x}) + \mathbf{eps}} \cdot s + m$$

where s, m, eps are fixed values. In our experiments, we take $s = 1, m = 0, eps = 10^{-7}$. The small value eps is added to avoid division by zero for those ECG records with constant value. After this transformation, ECGs will have mean m and standard deviation s (except for constant value ECGs).

To alleviate overfitting, data augmentation techniques including mixup [9], random masking [3] with zero values are adopted stochastically (with certain probability) for training the neural networks. Mixup performs convex linear combination of the training data as follows

$$\mathbf{x} = \alpha \mathbf{x_1} + (\mathbf{1} - \alpha)\mathbf{x_2},$$
$$\mathbf{y} = \alpha \mathbf{y_1} + (\mathbf{1} - \alpha)\mathbf{y_2},$$

where $0 < \alpha < 1$, $\mathbf{x_1}, \mathbf{x_2}$ are preprocessed ECGs, and $\mathbf{y_1}, \mathbf{y_2}$ are corresponding labels, and the input for the neural network is their convex linear combination $\mathbf{x}, \mathbf{y}$. No more augmentations, like random flip, are done, since they might completely change the interpretation of the standard-lead ECGs[1], as will be seen in Section 5. To suppress overconfidence which could help improving generalization capabilities of models, the technique of label-smoothing regularization (LSR) [10] is used. Let $\mathbf{y}$ be an one-hot label vector (the "hard" label), then LSR generates "soft" label vector via Equation (1)

$$\mathbf{y}' = (\mathbf{1} - \varepsilon)\mathbf{y} + \frac{\mathbf{1}}{\mathbf{K}}\varepsilon\mathbf{e}, \tag{1}$$

where K is the number of classes, $\mathbf{e}$ is the K-dimensional vector with all entries equaling one, and $\varepsilon \in [0, 1]$ is a weight factor. In our approach, we take $\varepsilon = 0.1$.

[1] which might not be the case for wearable dynamic ECGs

3. Neural Network Architectures

3.1. Convolutional Recurrent Neural Networks (CRNNs)

Inspired by previous work [3, 4, 11], we build our approach on top of a CRNN framework[2]. The philosophy is as follows.

CNNs consist of space translation equivariant convolution operators, which usually interleave with non-linear activations and downsampling operators capturing and fusing hierarchical local features. In our CRNN framework, CNNs serve as feature extractors (encoders) from raw input ECGs. In order to better modeling long-range dependency, optional (self-)attention modules (squeeze-and-excitation [12], global context [13], etc.) can follow or integrated in building block convolutions in the CNNs. An optional RNN can be added as well to make use of sequential information of the ECGs. Feature maps thus obtained are fed into multilayer perceptrons (MLPs, sequential linear layers) for ECG downstream tasks, including classification, sequence labeling (e.g. QRS complex detection [11]), etc.

3.2. Multi-branch CNNs

The most significant structures of ECGs are the P, Q, R, S, T waves and their rhythms which for example can be reflected by the sequence of wave intervals (RR intervals, PP intervals, QT intervals, etc.). Broadly speaking, these waves and intervals broadly have their "general" spectral characteristics which originate from the mechanism of the human heart's electric activities. Hence the receptive fields of the CNNs are crucial for ECG processing, or more widely for physiological signal processing [14]. Previous work [11] explicitly dealt with this point via multi-branch CNNs where each branch uses dilated convolutions [15] with different dilation factors. This CNN backbone is similar to the ASPP (Atrous Spatial Pyramid Pooling) head originally proposed in DeepLabV2[16], directly modelling the spectral characteristics. Therefore, we mainly experimented with multi-branch CNNs in our approach.

3.3. "Lead-wise" CNNs

The challenge [7] emphasises the utility of reduced-lead ECGs, hence in our approach we designed a "lead-wise" manner for the CNNs via grouped convolutions with number of groups divisible by the number of leads of the input ECGs. For example, 12 groups for the standard 12-lead ECGs. In this "lead-wise" settings, normalization layers are group normalizations [17] as well. In this way, CNNs extract features for each lead separately in parallel. Features from different leads are not fused until forwarded out from the CNNs. This provides the possibility to reuse parameters from the models trained on the standard 12-lead ECGs for reduced-lead ECGs, in which case one only needs to "fine-tune" the attention modules and the MLPs. This can play the role of general-purposed "backends" as in computer vision. Another advantage is that "lead-wise" CNNs are much smaller in the number of model parameters, with only slight drop of performance.

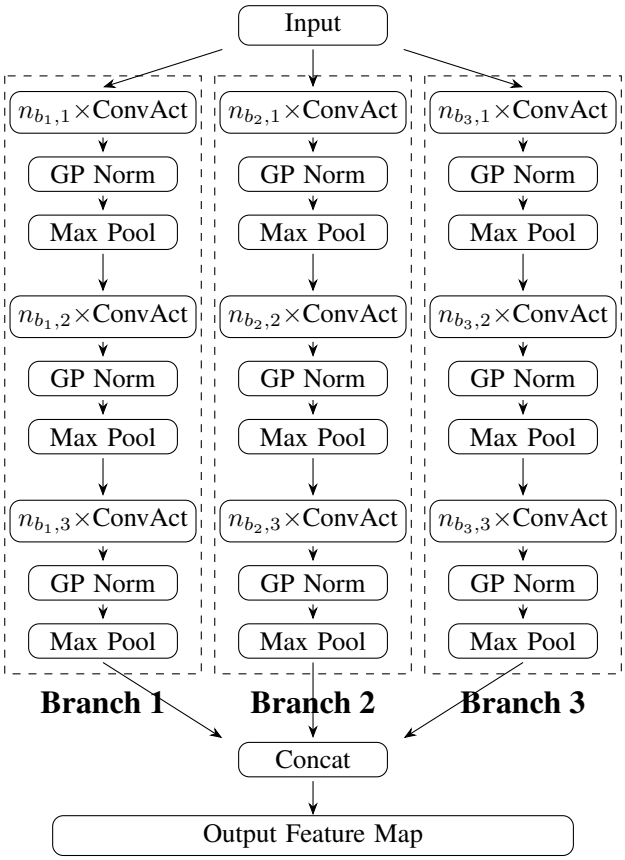


Figure 1: A typical 3-branch CNN. Abbreviations: "ConvAct" for grouped convolution layer followed by ReLU activation layer, "GP Norm" for group normalization layer, "Max Pool" for max pooling layer of kernel size 2. We set $(n_{b_i,1}, n_{b_i,2}, n_{b_i,3}) = (1, 2, 3)$, $i = 1, 2, 3$, in our challenge entry approaches. More details of the hyperparameters can be found in Table 1 and Table 2. This Architecture can shrink or expand horizontally by removing or adding branches, and shrink or expand vertically by removing or appending convolutions.

The major CNN architecture in our challenge entry approaches is plotted in Figure 1. The number of filters are

[2] indeed an ECG deep learning framework more broadly, available at https://github.com/DeepPSP/torch_ecg

# leads	12	6	4	3	2
$n_{b_i,1}\times$Conv	192	144	96	96	64
$n_{b_i,2}\times$Conv	384	288	192	192	128
$n_{b_i,3}\times$Conv	768	576	384	384	256

Table 1: Number of filters for the convolutions in the CNN described in Figure 1.

	Kernel Sizes	Dilations
Branch 1		$(1, 1, 1, 1, 1, 1)$
Branch 2	$(11, 7, 7, 5, 5, 5)$	$(2, 2, 4, 8, 8, 8)$
Branch 3		$(4, 4, 8, 16, 32, 64)$

Table 2: Kernel sizes and dilations for the convolutions in the CNN described in Figure 1.

listed in Table 1.[3] The number-of-leads-independent hyperparameters are gathered in Tabel 2. These hyperparameters are inherited from [11] directly. The whole network is gathered in Figure 2.

3.4. Variants of ResNet

For the purpose of conducting comparative studies, we experimented with various ResNet variants (e.g. [2, 4, 18, 19]) as CNN backbone for our neural networks. The original ResNet was proposed in [20] in 2016, and quickly became the most widely used neural network architecture. The TResNet [19] variant currently is still almost the state-of-the-art (SOTA) model for multi-label classification on the MS-COCO dataset [21]. Until recently, variants of ResNet [2,4] have long been dominating the research community of ECG processing using deep learning.

A typical architecture of ResNet consists of an input stem, followed by 4 stages, as depicted in Figure 3. The stem consists of convolutions with large kernel sizes and with [18–20] or without [4] down-sampling layers. The 4 stages consist of different numbers of stacked building blocks of basic type (Figure 4a) or of bottleneck type (Figure 4b). Attention mechanisms, which are aforementioned in Section 3.1, are usually integrated into these building blocks to further improve the performance of the networks.

[3]Mechanism of reuse of parameters for reduced-lead ECGs has not been established by the end of the challenge, hence we set this hyperparameter for each of the 5 lead sets.

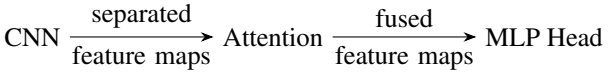


Figure 2: The whole network architecture. CNN is described in Figure 1. The attention module used in our approach is SENet with reduction ratio 8. Then adaptive max pooling is applied to the fused feature maps to reduce the number of channels to one. Finally, MLP consisting of one linear layer gives the predictions which are tensors of probabilities for each of the classes.

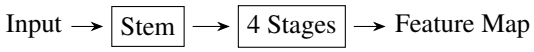
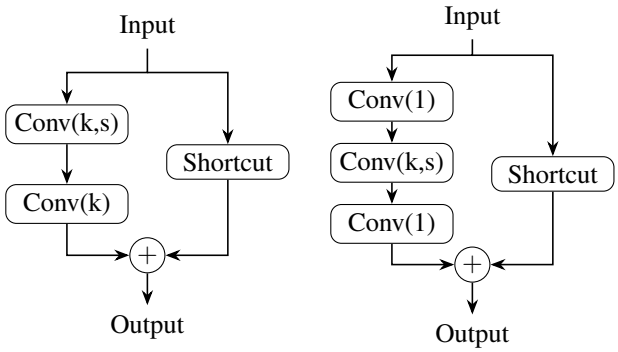


Figure 3: A typical architecture of ResNet. The Stem typically consists of convolutions with large kernel sizes to capture coarsely-level features and down-sampling layers to reduce computations. The 4 stages consists of stacks of units sketched in Figure 4 producing the feature maps for final classification.



(a) Basic Block

(b) Bottleneck Block

Figure 4: Sketch of 2 types of building blocks for ResNet. Conv(k,s) is convolution of kernel size k, and stride s; Conv(1) is convolution of kernel size 1 and stride 1. When $s = 1$, the shortcut is just the identity map; when $s > 1$, shortcut need to do down-sampling. When the main stream (the path passing from Input to Output through the convolutions) raises the number of channels, shortcut also needs to use convolutions of kernel size 1 to raise to the same number of channels. Other layers including activations, normalizations and dropouts are not plotted in this Figure. In (b), it is actually from the ResNet-B variant, rather than the original one.

The SE-variant (with squeeze-and-excitation attention integrated in the building blocks) of the ResNet model proposed in [4] will serve as the baseline for our comparative studied. This baseline will be denoted as `ResNet-NC-SE` ("NC" is short for "Nature Communications"). Its bottleneck variant, denoted as `ResNet-NC-BS`, is also experimented. Another series tested is the TResNet family [19]. TResNets adopted many new techniques for acceleration and/or improving prediction accuracy and robustness. These techniques include space-to-depth (S2D) [22] operation, anti-aliasing (AA) down-sampling [23], Since the smallest network (the `TResNet-M`) proposed therein has 21 building blocks,

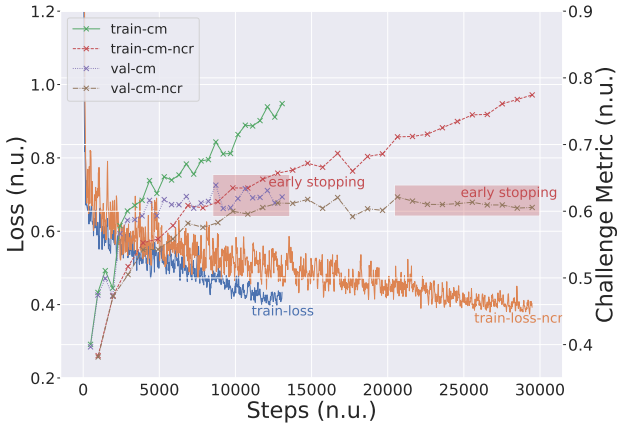


Figure 5: Two typical training processes using weighted BCE loss for neural networks with lead-wise branched CNN backbone. The suffix "ncr" refers to the challenge solution pipeline with **n**o **c**linical **r**ule based detector, in which case the whole 26 classes are included for training. "cm" is the abbreviation of the **c**hallenge **m**etric.

which is much too large for ECG processing, we designed and experimented 3 smaller variants, namely `TResNet-N` (nano), `TResNet-P` (pico), `TResNet-F` (femto). More details on architectures of the above networks are gathered in Table 3 and Table 4. Experiment results will be presented in Section 7.

4. Training Setups

The whole neural network development and validation work are done on a GPU server with 8×RTX3080. In most cases, we set batch size to be 64 and set the maximum number of epochs to be 50. 20% of the training data is left out for model evaluation and model selection. Early stopping is triggered if the challenge metric on the validation set does not grow for 8 epochs. Two typical training processes are depicted in Figure 5. In most cases throughout this paper, except for Section 6, when referring to "validation set", we mean this left-out set from the whole of the public training set.

4.1. Loss Functions

Since the challenge data is highly unbalanced, having a long tail distribution, we tested 2 loss functions aiming at dealing with such data distribution. Initially, we used weighted binary cross entropy (BCE) as the loss function. The weights are inversely proportional to the number of records of the classes. After studying other challengers' solutions, we found that the asymmetric loss [24] is more widely adopted. The asymmetric loss is defined as follows

$$ASL = \begin{cases} L_+ := (1-p)^{\gamma_+}\log(p), \\ L_- := (p_m)^{\gamma_-}\log(1-p_m), \end{cases}$$

where $p_m = \max(p-m, 0)$ is the so-called shifted probability, with probability margin m. The loss on one label of one sample is

$$L = -yL_+ - (1-y)L_-$$

By using this asymmetric loss, one is able to emphasize the contribution of positive samples or negative samples by adjusting the ratio of the focusing parameters γ_+ to γ_-. Typically in our experiments, γ_+ is fixed to be 0, and we set $\gamma_- = 0.2$. We observed augments of challenge metric by 0.03-0.05 on the validation set, which is very significant. More details can be found in Section 7.1. Therefore the asymmetric loss is our final choice of the loss function for training the neural network models.

4.2. Optimizers and Learning Rate Schedulers

Parameters of neural network models are optimized using the AMSGrad variant of the AdamW optimizer (denoted as `adamw_amsgrad`) [25]. For learning rate, which is the most important hyperparameter of an optimizer, we use the `OneCycle` scheduler [26] with maximum learning rate 0.002 and cosine annealing strategy to adjust learning rate during training. It was reported in the deep learning research community that the combination of `adamw_amsgrad` optimizer with `OneCycle` scheduler gives the best performance most of the time. For the purpose of comparative study, we also conduction experiments with constant learning rate 0.001. Corresponding results will be gathered in Section 7.1.

4.3. Model Inference and Model Selection

For model inference, to make binary predictions from probabilities, a threshold 0.5 is used. If none exceeds 0.5, then the class with the highest probability, as well as other classes with close enough (within a bias of 0.03) probabilities if any, are chosen as the binary output. The monitor for model selection is the challenge metric computed on the validation set.

5. Clinical Rule Based Detectors

Clinical rules based detectors are designed for the 5 ECG abnormalities listed in Section 2.1. From the authors' experiences of previous challenges and production

Network	Stem		Stage1		Stage2		Stage3		Stage4	
	type	chan	repeat, type	chan	repeat, type	chan	repeat, type	chan	repeat, type	chan
ResNet-NC-SE	Conv	64	1 Basic-SE	128	1 Basic-SE	192	1 Basic-SE	256	1 Basic-SE	320
ResNet-NC-BS	Conv	64	1 Bottle-SE	512	1 Bottle-SE	768	1 Bottle-SE	1024	1 Bottle-SE	1280
TResNet-N	S2D	56	2 Basic-SE	56	2 Basic-SE	112	2 Bottle-SE*	896	2 Bottle	1792
TResNet-P	S2D	56	1 Basic-SE	56	1 Basic-SE	112	1 Bottle-SE*	896	1 Bottle	1792
TResNet-F**	S2D	32	1 Basic-SE	32	1 Basic-SE	64	1 Bottle-SE*	512	1 Bottle	1024

Table 3: Architectures of Variants of ResNet. Types and number of repeats and output channels of the building blocks used are listed. For the stem, "Conv" means one convolutional layer with kernel size 17 and without down-sampling; "S2D" refers to the space-to-depth operation, possibly followed by a convolution with kernel size 1 to match the number of output channels of the stem.
* SE module follows the 2nd convolution of the block, the rest follows the last convolution of the block.
** Convolutions are separable convolutions.

Network	Kernel Size		Down-Sampling	
	Stem	4 Stages	Stem	4 Stages
ResNet-NC-SE	17	17	1	$(4, 4, 4, 4)$
ResNet-NC-BS	17	17	1	$(4, 4, 4, 4)$
TResNet-N	1	15	4	$(1, 2, 2, 2)$
TResNet-P	1	17	4	$(1, 2, 2, 2)$
TResNet-F	1	17	4	$(1, 2, 2, 2)$

Table 4: Details of the kernel sizes and down-sampling ratios of the stem and the building blocks of the 4 stages in the ResNet variants.

systems, post-processing using clinical rules is an excellent supplement to machine learning models. Details of these detectors are as follows:

1. "Brady" (bradycardia): average heart rate ≤ 60 BPM (beat per minute) or equivalently average RR-intervals ≥ 1 second.
2. "LAD" (left axis deviation) and "RAD" (right axis deviation): positivity checking of QRS complexes of leads I, aVF ("2-lead" method) or of leads I, II, aVF ("3-lead" method) as in [27]. More precisely
   - "2-lead" method:
   – "LAD": lead I is positive; lead aVF is negative;
   – "RAD": lead I is negative; lead aVF is positive.
   - "3-lead" method:
   – "LAD": lead I is positive; lead II, aVF are negative;
   – "RAD": lead I is negative; lead II, aVF are positive.

   "2-lead" method is simpler, but might have false positives.
3. "LQRSV" (low QRS voltage): peak-to-peak amplitudes of more than 80% of the QRS complexes are ≤ 0.5 mV in the limb leads (I, II, III, aVR, aVL, aVF), or ≤ 1 mV in the precordial leads (V1-V6). If R peak detection fails, amplitude check will be done within sliding windows of length 0.12 second.
4. "PR" (pacing rhythm): raw ECGs are high-pass filtered with cutoff frequency 47 Hz, and spike (peak) detection with prominence threshold of 0.3 follows.

Detection of the first 4 abnormalities relies heavily on R peak detection, for which we use the function `xqrs_detect` from the WFDB package [5, 28] for simplicity. This function however is far from optimal, causing nonnegligible amount of miss-classifications. The 5-dimensional outputs and the 21-dimensional outputs from DNNs are naively merged to produce the final predictions.

It can be inferred from the clinical rules that a small proportion of ECG arrhythmias can not be reasonably detected under certain combination of reduced leads, which would be further discussed in Section 8.

6. Challenge Results

Our challenge entries for CinC2021 mainly uses two configurations, namely the proposed hybrid method, and the pure DNN approach. Best scores of challenge entry submissions and offline experiments are gathered in Table 5.

We carried out offline expriments using CNNs without the "lead-wise" setting, but no successful entry submission was made, perhaps due to the increase of model size that exceed the computation capacity of the challenge official computing environment. These ablation studies, along with ablation studies focusing on other aspects, are reported in Section 7.

7. Ablation Study and Beyond

7.1. Effect of Loss Function, Normalization, Augmentations, and More

As is already mentioned in Section 4.1, we find that the asymmetric loss largely improves the performance of neural network models. We are going to show this point in this section. We also conduct various experiments on other

Leads	Training	Validation	Test	Ranking
12	0.62	0.51	NA	NA
6	0.59	0.47	0.33	NA
4	0.60	0.47	0.35	NA
3	0.61	0.48	0.33	NA
2	0.59	0.48	0.33	NA
12-cr	0.64	0.51	NA	NA
6-cr	0.61	0.49	NA	NA
4-cr	0.61	0.44	NA	NA
3-cr	0.61	0.46	NA	NA
2-cr	0.59	0.43	NA	NA

Table 5: Challenge scores (top 5 rows) for the final entry. The final entry used the "no clinical rule" pure DNN approach. It failed on the 12-lead test set (more exactly the 12-lead "UMich test" set). The auxiliary bottom 5 rows (with "-cr" suffix) describes performances of our hybrid approach mixing DNN and clinical rules. Scores in the "Training" column are typical scores on the 20% left-out train-validation data, as described in Section 4.

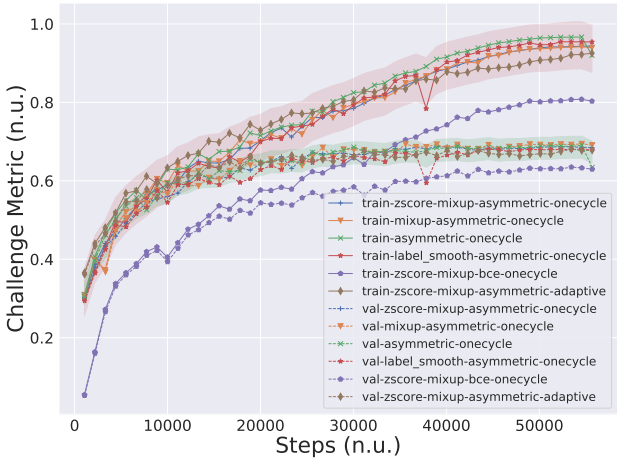


Figure 6: Ablation Study 0: Challenge scores on both the training and validation sets. The techniques used are concatenated by "-" in the legend of the figure.

techniques including normalization, augmentations, learning rate schedulers, etc., aiming at figuring out whether they truly help augmenting model performances or not. This servers as a part of our ablation studies in a broader sense, which we call Ablation Study 0.

We list the choices for the components for training neural networks that are altered in this part of ablation study:

- Loss functions: asymmetric loss, weighted BCE loss;
- Normalization: z-score, no normalization;
- Augmentations: mixup, label smoothing;
- Learning rate scheduler: `OneCycle` scheduler, self-adaptive learning rate by `adamw_amsgrad` with initial value 0.001;

The baseline model for conducting this part of ablation study is the CRNN model with the following (sequential) components:

- CNN: `ResNet-NC-SE`;
- RNN: stack of 2 bidirectional LSTM layers with hidden size 192;
- Attention module: SE module with reduction ratio 8;
- MLP head: stack of 2 linear layers with 1024 intermediate features.

Note that in [4], the feature map from the CNN backbone are flattened (which can be called "depth to space") to feed into a linear layer for final prediction, while we just use a global max pooling layer before the linear layers.

Figure 6 demonstrates the growth of the challenge metric on both the training and the validation set. Mini-batch loss against the number of steps is plotted in Figure 7. One can draw the conclusion that the asymmetric loss contributes most to the improvement of the baseline model. The `OneCycle` learning rate scheduler takes the second

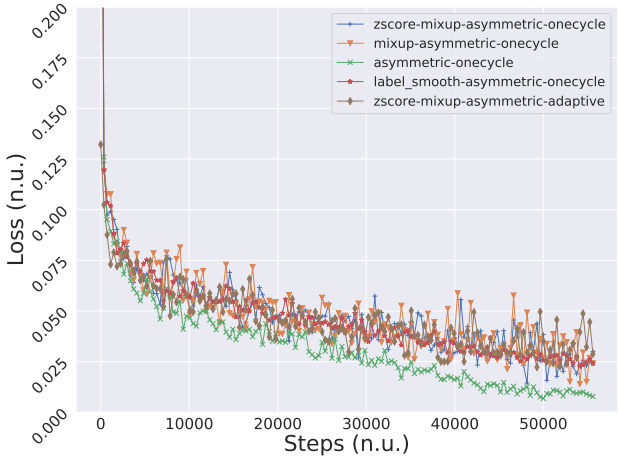


Figure 7: Ablation Study 0: (Mean value of) Mini-batch asymmetric loss on the training set. The experiment using the weighted BCE loss is not included, since it's meaningless to compare it with the rest 4. The curves are smoothed using exponential moving average with weight 0.6.

place but the augment is much less significant. Influences of other techniques are small. Neural network trained with mixup augmentation performs the best on the validation set. The loss curves on the training set with augmentation techniques have larger oscillations, while the loss curve without any augmentation techniques (the green "asymmetric-onecyle" curve with cross markers in Figure 7) decreases more stably.

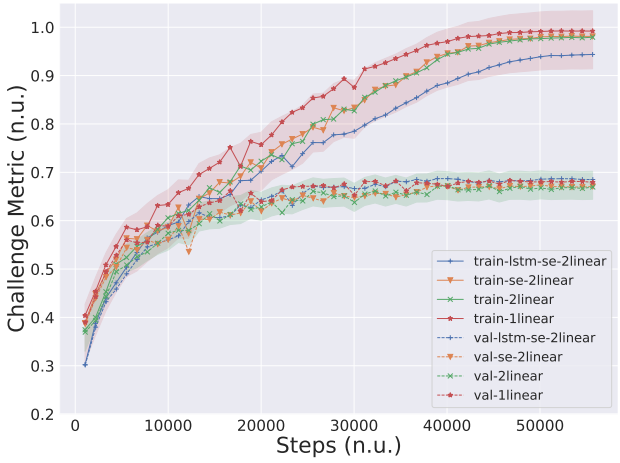


Figure 8: Ablation Study 1: Curves of challenge scores on both the training and validation sets. Names of neural network components other than the `ResNet-NC-SE` backbone are concatenated by "-" in the legend.

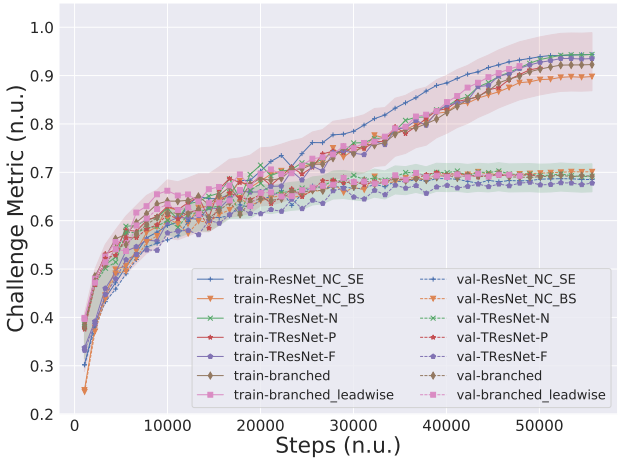


Figure 9: Ablation Study 2: Curves of challenge scores on both the training and validation sets from neural networks composed of different CNN backbones followed by LSTM module, SE module, and two linear layers.

7.2. Effect of Neural Network Complexity

Intuitively, increasing the network complexity helps improving model performances. However, it is not always true, sometimes even the opposite. We designed the following sets of experiments to check the contributions of various component of the whole network to the challenge task.

1. Ablation Study 1: we fix the CNN backbone `ResNet-NC-SE`, and gradually shrink the other parts:
   - LSTM + SE attention layer + 2 linear layers (denoted `lstm-se-2linear`);
   - SE attention layer + 2 linear layers (denoted `se-2linear`);
   - 2 linear layers (denoted `2linear`);
   - 1 linear layer (denoted `1linear`).
2. Ablation Study 2: we change the CNN backbones, and fix the rest part of the network to be LSTM + SE attention layer + 2 linear layers. The CNN backbones includes the ResNet variants introduced in Section 3.4, as well as the multi-branched CNN introduce in Section 3.2 and its lead-wise variant introduce in Section 3.3.
3. Ablation Study 3: this ablation study has similar setting as in Ablation 2, the difference is that the rest part of the network other than the CNN backbone is changed to the simplest single linear layer.
4. Ablation Study 4: we use the same baseline model as in Ablation Study 0 to test on the 5 lead-sets listed in Section 1. As the results on the reduced-lead ECGs using our novel architectures have already been presented in 6, we here use this baseline model in this ablation study.

One can observe in Figure 8 that the 4 listed in Ablation Study 1 have very close performances on the validation set. However, the extra LSTM layer largely suppresses the overfitting on the training set. The other 2 combinations (`se-2linear` and `2linear`) with extra components other than one bare linear layer have lowered performances. Increasing the network complexity does not always result in better performances.

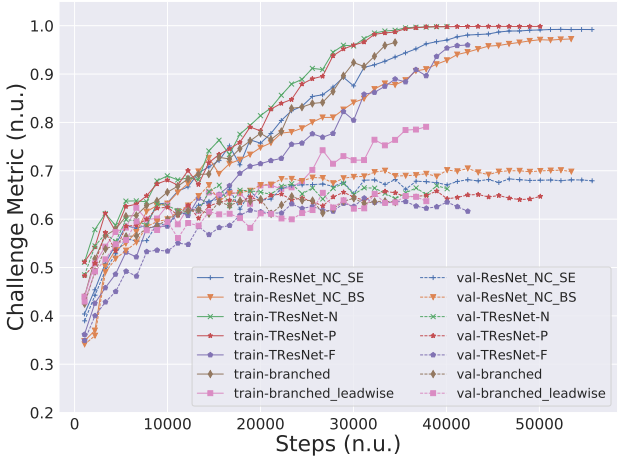


Figure 10: Ablation Study 3: Curves of challenge scores on both the training and validation sets from neural networks composed of different CNN backbones followed by only one linear layers.

With the help of extra components (`lstm-se-2linear`), neural networks with different CNN backbones also have very close performances as illustrated in Figure 9. The gap between the challenge metrics on the training set and on the validation set is significantly smaller when using the `ResNet-NC-BS` backbone. The bottleneck building block has the effect of regularization to some extent.

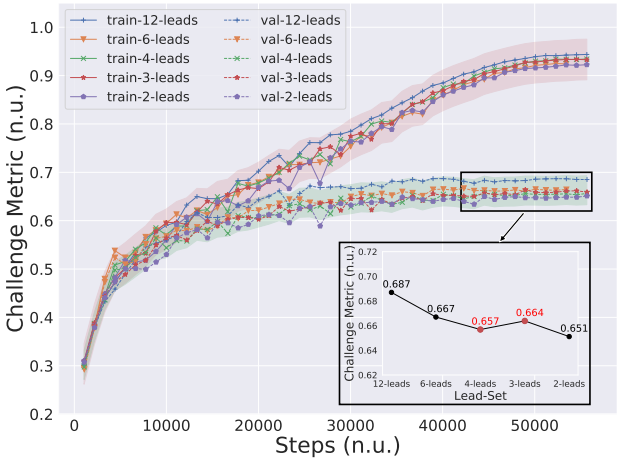


Figure 11: Ablation Study 4: curves of Challenge scores on both the training and validation set from the baseline model tested on the 5 lead-sets.

When CNN backbones are concatenated with only a bare linear layer for prediction, performances of the networks diverge, as can be observed in Figure 10. Networks completely without any extra structures (inner SE, bottleneck, etc.) degrade dramatically. Other networks with higher complexity overfit very fast on the training set.

From Figure 11, we find that when the neural network has adequate complexity, the drop from using all the 12 leads to using only 2 of them for making ECG arrhythmia classifications is very slight (only 0.036 in terms of challenge metric) and is reasonably acceptable. As already been observed in Table 5, another interesting phenomenon is that the model performance is higher on the 3-lead-set (I, II, V2) than on the 4-lead-set (I, II, III, V2), although the former is a subset of the latter. The extra information from lead III "poisons" the neural networks.

The overall statistics of all the ablation studies are gathered in Table 6.

8. Discussion and Conclusions

The hybrid method of DNNs and clinical rules proposed in this paper offers an effective approach for automated auxiliary multi-lead ECG diagnosis systems, providing a balance between performance and interpretability. It can be inferred from the results that reduced-lead ECGs, even 2-lead ECGs in the extreme case, provide sufficient information for making reliable auxiliary diagnoses, with performances (challenge score) only slightly dropped by at most 0.03, compared to the standard 12-lead ECGs on the validation set. The neural networks are validated on a wide range of cardiac abnormalities, suggesting that they are capable to assist the cardiologist in real-world applications.

There are limitations and left for future work. First, the multi-branch CNNs for feature extraction are not optimal, being inferior to some of the ResNet variants. Its structures and hyperparameters both have to be optimized. Despite the ablation studies presented in Section 7, a more thorough search for more effective architectures should and is undertaken by the authors using the ECG deep learning framework mentioned in Section 3.

Second, it is observed in Table 5 that performances of the hybrid entries on the reduced 4-lead, 3-lead and 2-lead ECGs dropped slightly larger than pure DNN entries. The hyperparameters of clinical rule based detectors are set empirically, which should be optimized via grid searches.

Label heterogeneity and insufficiency across datasets should also be noted. We observed labels that violates clinical criteria. For example, some "LAD" records violates the "3-lead" method which is more exact than the "2-lead" method mentioned in Section 5. An example is illustrated in Figure 12.

Most importantly, the mechanism of parameters reuse is to be further established, in order to fully utilize flexible light weight solutions to reduced-lead ECGs provided by the "lead-wise" CNNs.

Acknowledgments

The authors would like to thank professor Deren Han from LMIB, School of Mathematical Sciences, Beihang University and professor Wenjian Yu from the Department of Computer Science and Technology, BNRist, Tsinghua University for generously providing GPU servers to help accomplish this work.

References

[1] Virani SS, Alonso A, Aparicio HJ, Benjamin EJ, Bittencourt MS, Callaway CW, et al. Heart Disease and Stroke Statistics – 2021 Update: a Report from the American Heart Association. Circulation 2021;143(8):e254–e743.

[2] Hannun AY, Rajpurkar P, Haghpanahi M, Tison GH, Bourn C, Turakhia MP, et al. Cardiologist-level Arrhythmia Detection and Classification in Ambulatory Electrocardiograms using a Deep Neural Network. Nature Medicine 2019; 25(1):65.

[3] Yao Q, Wang R, Fan X, Liu J, Li Y. Multi-class Arrhythmia Detection from 12-lead Varied-length ECG Using Attention-based Time-Incremental Convolutional Neural Network. Information Fusion 2020;53:174–182.

[4] Ribeiro AH, Ribeiro MH, Paixão GM, Oliveira DM, Gomes PR, Canazart JA, et al. Automatic Diagnosis of the 12-lead ECG using a Deep Neural Network. Nature Communications 2020;11(1):1–9.

[5] Goldberger AL, Amaral LA, Glass L, Hausdorff JM, Ivanov PC, Mark RG, et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation 2000;101(23):e215–e220.

CNN backbone	# leads	LSTM	SE	# linear	# params	Best Score	Params Efficiency	Training Speed (sig/s)	Inference Speed (sig/s)
ResNet-NC-SE	12	×	×	1	6.79 M	0.683	10.1	399	2984
	12	×	×	2	7.14 M	0.673	9.42	391	2973
	12	×	✓	2	7.14 M	0.674	9.41	386	2640
	12	✓	✓	2	8.92 M	0.687	7.70	352	2811
	6	✓	✓	2	8.91 M	0.667	7.48	365	3070
	4	✓	✓	2	8.91 M	0.657	7.31	387	3092
	3	✓	✓	2	8.91 M	0.664	7.45	392	3153
	2	✓	✓	2	8.91 M	0.651	7.37	401	3171
ResNet-NC-BS	12	×	×	1	8.75 M	**0.705**	8.06	353	2037
	12	✓	✓	2	12.3 M	0.701	5.69	300	1974
TResNet-N	12	×	×	1	14.0 M	0.674	4.81	244	1395
	12	✓	✓	2	18.4 M	**0.703**	3.83	182	1139
TResNet-P	12	×	×	1	8.10 M	0.658	5.62	378	2046
	12	✓	✓	2	12.5 M	**0.700**	8.12	252	1553
TResNet-F	12	×	×	1	**1.23 M**	0.643	**52.3**	373	**3325**
	12	✓	✓	2	4.42 M	0.679	15.4	262	1785
Branched	12	×	×	1	6.77 M	0.647	9.56	**408**	515
	12	✓	✓	2	10.2 M	**0.698**	6.88	146	362
Branched-LW	12	×	×	1	2.33 M	0.654	28.0	357	536
	12	✓	✓	2	7.46 M	**0.699**	9.37	134	360

Table 6: Overall statistics of Ablation Study 1, 2, 3, 4. Params Efficiency is computed as $10^8 \times (\text{Best Score})/(\#\text{ Params})$. Training speed is computed under the settings stated in Section 4. Inference speed is computed using one RTX3080 based on a single run on the training set.

Figure 12: Plot of the first 6 seconds of leads I, II, aVF of record "HR05631" from the PTB-XL subset. It has "LAD" in its scored label list. Leads I and II are positive, lead aVF is negative. It will be classified as "LAD" by the "2-lead" method detector proposed in Section 5, while classified as non-"LAD" by the "3-lead" method detector.

[6] Perez Alday EA, Gu A, Shah A, Robichaux C, Wong AKI, Liu C, et al. Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020. Physiological Measurement Nov 2020;41. Doi: 10.1088/1361-6579/abc960.

[7] Reyna MA, Sadr N, Perez Alday EA, Gu A, Shah A, Ro-

bichaux C, et al. Will Two Do? Varying Dimensions in Electrocardiography: the PhysioNet/Computing in Cardiology Challenge 2021. Computing in Cardiology 2021;48:1–4.

[8] Tihonenko V, Khaustov A, Ivanov S, Rivin A, Yakushenko E. St Petersburg INCART 12-lead Arrhythmia Database. PhysioBank PhysioToolkit and PhysioNet 2008;Doi: 10.13026/C2V88N.

[9] Zhang H, Cisse M, Dauphin YN, Lopez-Paz D. mixup: Beyond Empirical Risk Minimization. International Conference on Learning Representations 2018;URL https://openreview.net/forum?id=r1Ddp1-Rb.

[10] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016; 2818–2826.

[11] Cai W, Hu D. QRS Complex Detection Using Novel Deep Learning Neural Networks. IEEE Access 2020;8:97082–97089.

[12] Hu J, Shen L, Albanie S, Sun G, Wu E. Squeeze-and-Excitation Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 2020;42(8):2011–2023.

[13] Cao Y, Xu J, Lin S, Wei F, Hu H. GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond. In 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW). 2019; 1971–1980. Doi: 10.1109/ICCVW.2019.00246.

[14] Baek S, Jang J, Yoon S. End-to-End Blood Pressure Prediction via Fully Convolutional Networks. IEEE Access 2019; 7:185458–185468.

[15] Yu F, Koltun V. Multi-Scale Context Aggregation by Dilated Convolutions. In ICLR. 2016; .

[16] Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. IEEE Transactions on Pattern Analysis Machine Intelligence 2018;40(04):834–848.

[17] Wu Y, He K. Group Normalization. International Journal of Computer Vision 7 2019;128(3):742–755. Doi: 10.1007/s11263-019-01198-w.

[18] He T, Zhang Z, Zhang H, Zhang Z, Xie J, Li M. Bag of Tricks for Image Classification with Convolutional Neural Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019; 558–567.

[19] Ridnik T, Lawen H, Noy A, Ben Baruch E, Sharir G, Friedman I. TResNet: High Performance GPU-Dedicated Architecture. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021; 1400–1409.

[20] He K, Zhang X, Ren S, Sun J. Deep Residual Learning for Image Recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 6 2016; 770–778.

[21] Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft COCO: Common Objects in Context. In European conference on computer vision. Springer, 2014; 740–755.

[22] Sandler M, Baccash J, Zhmoginov A, Howard A. Non-Discriminative Data or Weak Model? On the Relative Importance of Data and Model Resolution. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops. 2019; 0–0.

[23] Zhang R. Making Convolutional Networks Shift-Invariant Again. In International conference on machine learning. PMLR, 2019; 7324–7334.

[24] Ridnik T, Ben-Baruch E, Zamir N, Noy A, Friedman I, Protter M, et al. Asymmetric loss for multi-label classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021; 82–91.

[25] Reddi SJ, Kale S, Kumar S. On the Convergence of Adam and Beyond. In 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada. OpenReview.net, 2018; .

[26] Smith LN, Topin N. Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates. In Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, volume 11006. International Society for Optics and Photonics, 2019; 1100612.

[27] Kashou AH, Basit H, Chhabra L. Electrical Right and Left Axis Deviation. StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing, Jan 2021. Available from: https://www.ncbi.nlm.nih.gov/books/NBK470532/.

[28] Xie C, McCullum L, Johnson A, Pollard T, Gow B, Moody B. Waveform Database Software Package (WFDB) for Python, 2021. Doi: 10.13026/G35G-C061.