ECG R-peak detection – CNN and HMM approach

A Course Project Report for E511 – Machine Learning for Signal Processing, IU Bloomington

By Salil Harsulkar (samihars@iu.edu) and Yanting Wan (yanwan@iu.edu)

December 14, 2018

1. Abstraction

Detecting R-peak is fundamental task in monitoring the electrocardiogram (ECG), so decades of studies have contributed to it and achieved great performance. However, it remains to be a challenge when ECG is captured in noisy environments. To detect R-peak in ECG with noise, traditional methods design several filters, and pick parameters manually to eliminate the falsely detected R-peak caused by noise. In this paper we study two state-of-the-art models: One is combining Convolutional Neural Networks (CNNs) with multi-layer perceptron (MLP). The purpose of using this model is to filter and learn the representation of raw ECG automatically unlike traditional methods that manually preprocess raw ECG. The other is Hidden Markov Model (HMM), which models the transition probability of state change and state output to suit the underlying data sequence and observed values. As ECG signal is basically time series data with its wave segments occurring in a predefined order where one segment is only dependent on the occurrence of its previous segment, HMMs are naturally suitable to describe the ECG signal. In this work, we show the comparison of performance of the two models for the task of detecting R-peaks on MIT-BIH Arrhythmia Database [16].

2. Introduction and Prior Work:

Electrocardiogram (ECG) is a graphical representation of electrical activity of the heart and has been commonly used for cardiovascular disease diagnosis. A typical ECG-based heartbeat mainly consists of three waves including P-wave, QRS complex, and T-wave. The QRS complex is the most prominent feature and R peak is the peak in QRS complex. QRS complex can be used to obtain additional useful clinical information from ECG signals, such as RR interval, QT interval, and PR interval, etc. Thus, R-peak detection is critical for ECG-based health evaluation. Abnormal values of R-R interval are indicative of arrythmia and sometimes of a medical condition called Premature Ventricular Contraction. In this paper we study two state-of-the-art models: One is combining Convolutional Neural Networks (CNNs) with multi-layer perceptron (MLP), the other is Hidden Markov Models.

2.1 CNN Model

With the advent of deep neural networks (DNNs), lots of improvements have been made in recent years in well-established pattern recognition areas, such as images and speeches. In the meantime, various neural network architectures have been proposed, such as CNNs and Long Short-Term Memory (LSTM) models. However, almost all works that detect R peaks make use of traditional signal processing techniques: linear filtering, Hilbert Transform, wavelet transform, which exploits the characteristic of QRS complex and have shown nearly perfect performance on experimental dataset.

Recently, a trend in the machine learning community has emerged towards deriving a representation of the input signal directly from raw, unprocessed data with the notion that the network can learn the intermediate representation of raw input signal and the acquired representation suits the given task better. Motivated by this idea, we treat ECG as a graphical representation of heart beating, develops a CNN to represent the feature of ECG segment graph.

2.1.1 Related Work

In one of traditional R-peak detection methods that has great performance, Zhu and Dong [12] built a R-peak detection program based on Shannon energy envelop. There are three steps in their program: The first step is applying Band-pass filtering (BPF), differencing and Amplitude Transform (AT). In this step, three parameters were selected: cutoff frequencies (5 and 18 Hz), 6th-order Chebyshev type I filter, and the manually picking value that differenced result is divided by in AT. The second step is calculating Shannon energy of the output in the 1st step, then applying a Low Pass filter. In this step, two parameters were selected: cutoff frequency and type of low pass filter (moving average). The last step is peak detection (PD), False-R Detection (FRD) and False Noise Detection (FND). In FRD, threshold is manually selected to remove false R-peak and in FND, time-interval length is manually chosen to bring back some removed peaks. Although this program achieved great result in the collected dataset, but it requires too many preprocessing steps and those manually selected parameters in each step are only suitable for the collected dataset because they are derived from dataset. Therefore, the program is not only complex but also can't generalize to other data. To make R-peak detection program less complex and generalize well, we used DNNs in our proposed model.

In one of the first studies that use few preprocessing techniques and learn features of ECG automatically, Xiang, Lin and Meng [15] proposed a CNN+MLP model to detect R-peak, and experiments on MIT-BIH Arrhythmia Database show that they got nearly perfect performance. However, in their paper, they don't mention the configuration of CNN and MLP, which leaves space for us to experiment.

2.1.2 Contribution (Novelty of this work):

CNN does a great job in capturing the feature of image. In this work, we treat ECG as a graphical representation of the electric activity of heart beat, and R-peaks as abnormal points in the graph. Thus, we use CNN to capture the feature of "the graph of heart", then classifies each feature vector into normal (non-R-peak) and abnormal (R-peak) points, which is different from viewing ECG as time series.

2.2 Hidden Markov Models:

2.2.1 Introduction:

Hidden Markov Models (HMM) are stochastic models that tune the probability of state transition and state emissions to suit the underlying data sequence and observed values. As ECG signal is basically time series data where its wave segments (P, QRS, T waves) only depend on the occurrence of its previous segment and only occur in a known order, HMMs are naturally suitable to describe them. All studies and research papers we surveyed unanimously conclude that HMM based approaches are best suited for ECG segmentation and are powerful enough to provide high classification accuracy comparable with the state-of-the-art techniques used on ground today.

2.2.2 Related Work:

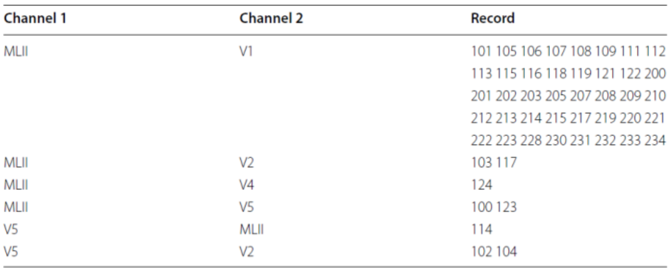
The scientific literature that we surveyed describing the application of HMM for ECG analysis can be classified into three main categories, namely, a. beat detection, b. wave segmentation, and c. arrythmia classification. Coast et all [6] were the first to apply HMM based approach for arrythmia classification in 1990 which showed promising results. The next two decades saw further improvements in the applications of wave segmentation and beat detection using better modelling of the state transitions [2], improving on feature extraction methods [4], [8] and better HMM architecture modelling [3], [5]. [1], [7] provides a detailed survey of different machine learning methods employed for automated analysis of ECG. There are two main approaches found in available literature for feature modelling – a. using wavelet transforms and, b. dimensionality reduction. Andreão et. all [4] studied the effect of different wavelets on the accuracy of HMM classifier for wave segmentation. They concluded that out of the six different wavelet functions studied, the Mexican Hat wavelet gives the best performance. On the other hand, Frénay et. all [2] found that Coiflet wavelet with two dyadic scales worked considerably well. Kaya et all in [8] explore Principal Component Analysis and Independent Component Analysis on the statistical and temporal features extracted from ECG signal. Using different classifiers for the application of QRS detection, the accuracy was consistently above 95%. Rodríguez et all explored PCA with Hilbert transform and adaptive filtering technique for QRS detection resulting in similar accuracy.

2.2.3 Contribution (Novelty of this paper):

Although the literature we surveyed for HMM heavily focused on ECG wave segmentation, specifically detecting R-wave using a two state HMM model has not been a subject of any study yet. The training and testing strategy and interpretation of the evaluation metrics was also developed independent of any support from available literature.

3. Data Introduction:

R-Peak Detection is investigated on ECG data by using the MIT-BIH Arrhythmia (MIT-BIH-AR) database which contains 48 ECG recordings from 47 subjects, which are studied by the BIH Arrhythmia Laboratory and each recording is sampled at 360 Hz for 30 min with 11-bit resolution over a 10mV range. Two or more cardiologists worked independently to annotate each record for R-Peak position; and disagreements were resolved to obtain computer readable annotation files which are included in this dataset. Each recording contains two ECG leads, one lead is modified-lead II (MLII) and the other lead is mainly lead V1, sometimes V2, V4 or V5. In our study, because MLII yields high-amplitude normal QRS complexes in most subjects, and others usually offers a nearly optimal frontal-plane projection of any ectopic beats that happen to be of low amplitude in MLII, only MLII is used in our study to emphasize QRS complexes. Thus, two recordings,102 and 104, are discarded in our study because they don't contain MLII.



Channel 1	Channel 2	Record

MLII	V1	101 105 106 107 108 109 111 112
		113 115 116 118 119 121 122 200
		201 202 203 205 207 208 209 210
		212 213 214 215 217 219 220 221
		222 223 228 230 231 232 233 234

MLII	V2	103 117
MLII	V4	124
MLII	V5	100 123
V5	MLII	114
V5	V2	102 104

Table 1: MIT-BIH Recordings and its corresponding channels

4. **Evaluation Metrics:**

Two measures are usually considered for evaluating the performance of heartbeat segmentation: sensitivity and positive predictivity, which are defined as: $\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$ $\text{Positive predictivity} = \frac{\text{TP}}{\text{TP} + \text{FP}}$

where TP (True Positive) indicate the number of R-waves correctly identified, FP (False Positive) indicate number of segmentations that do not correspond to the R-waves, and FN (False Negative) indicate number of segmentations that were not performed, respectively. [7]

5. **R-peak detection - Anomaly detection approach using CNN:**

5.1 Model Design

Input: We segment the differenced ECG to 0.833s long sequences after we preprocess the time-sequences to have zero mean and unit variance to take different voltage level of ECG among subjects. The reason of applying difference is emphasizing QRS complex and the reason that choosing 0.833s is 0.833s covers the period of heart beat. At 360 Hz sampling rate, this corresponds to a 300-dimensional input vector.

Temporal Convolution: We use 5 channels of filters with a 0.1s window in order to extract fine-grained features of ECG. At 360 Hz, this corresponds to a sequence with length 36. 20 channels of filters equal to 20 1-d convolutions, and 1-d convolution is a key component of our model. Here is the formula of 1-d convolution that operate on discrete-time signal h(k):

$$(f \star h)(t) = \sum_{k=-T}^{T} f(t) \cdot h(t-k)$$

Where f(t) is the filter (window) that will be learned from our data.

Activation: After being applied to temporal convolution, the normalized differenced ECG is activated by Rectified Linear Unit (ReLu). ReLu is defined as: y = max(x,0).

Pooling: A subsampling process. We perform a max-pooling across the time domain of activated sequence with a pool size of 2, which can be viewed as down-sampling to 180 Hz.

MLP: After features are extracted by CNN, we use MLP with two hidden layers with 64 neurons each. The output will be between 0 and 1, which indicates the likelihood of the current detection point being R-peak.

To train the network, we use backpropagation scheme with cross entropy as objective function.

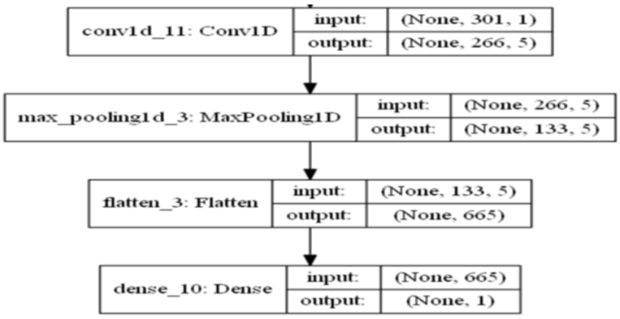


Figure 1: Configuration of CNN model

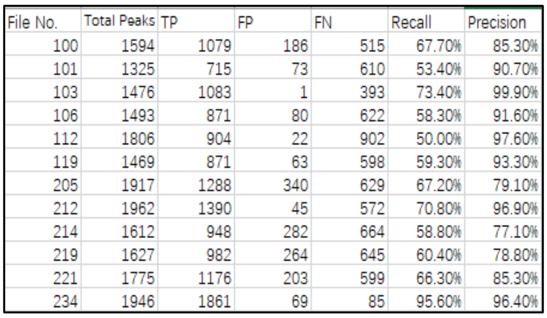
5.2 Experiment Setup

For each ECG recording, the first 70% of the recording is used as training set and the rest 30% is used as test set. To test whether a given detection point in raw ECG is a R-peak, following the preprocessing step in training step: differencing raw ECG, then normalizing the differenced ECG, adding the neighboring 300 points of the detection point to form a vector, finally feeding the vector into model to output whether it's a R-peak. To test generalization of the model, we singled recording 200 out because recording 200 recorded the ECG of a patient that is not covered in both training and test set.

5.3 Result

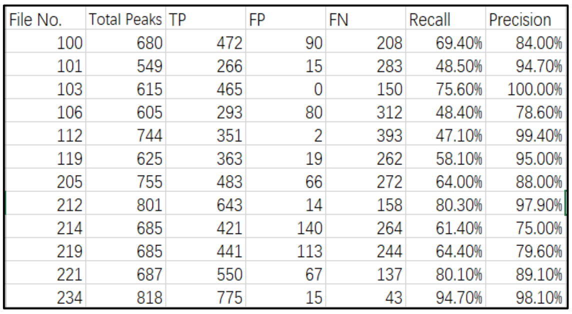
In section 4, we discussed the two metrics that evaluate model: Sensitivity and Positive predictivity, so we used them as results to show how well the CNN model performs. The overall training sensitivity ("Recall" in the table) is 65.1% and training precision (positive predictivity) is 89.3%, and the overall test sensitivity is 66% and test precision is 89.95%. Notice that the performance of model on training set is very similar to that on test set, which makes sense because the recordings are from the same dataset, thus being modified by same techniques. Then testing the model on recording 200, we get sensitivity 61% and positive predictivity 88%. Notice that there is more than 20% gap between sensitivity and positive predictivity in every performance test, we will discuss the reason of it in "Discussion" section.

The two table below shows the training and test performance of samples of recordings:



File No.	Total Peaks	TP	FP	FN	Recall	Precision
100	1594	1079	186	515	67.70%	85.30%
101	1325	715	73	610	53.40%	90.70%
103	1476	1083	1	393	73.40%	99.90%
106	1493	871	80	622	58.30%	91.60%
112	1806	904	22	902	50.00%	97.60%
119	1469	871	63	598	59.30%	93.30%
205	1917	1288	340	629	67.20%	79.10%
212	1962	1390	45	572	70.80%	96.90%
214	1612	948	282	664	58.80%	77.10%
219	1627	982	264	645	60.40%	78.80%
221	1775	1176	203	599	66.30%	85.30%
234	1946	1861	69	85	95.60%	96.40%

Table 2: Samples of training result



File No.	Total Peaks	TP	FP	FN	Recall	Precision
100	680	472	90	208	69.40%	84.00%
101	549	266	15	283	48.50%	94.70%
103	615	465	0	150	75.60%	100.00%
106	605	293	80	312	48.40%	78.60%
112	744	351	2	393	47.10%	99.40%
119	625	363	19	262	58.10%	95.00%
205	755	483	66	272	64.00%	88.00%
212	801	643	14	158	80.30%	97.90%
214	685	421	140	264	61.40%	75.00%
219	685	441	113	244	64.40%	79.60%
221	687	550	67	137	80.10%	89.10%
234	818	775	15	43	94.70%	98.10%

Table 3: Samples of test result

6. R-peak detection - Binary state classification using HMM:

6.1 Model design:

As the aim of this effort is to detect QRS segment, we divide the ECG signal into two states namely,

State 0 – No R-wave which comprises of the baseline and P, T and U waves, and

State 1 – QRS complex.

This translates our problem into a binary classification problem splitting the ECG wave into two obviously separable parts. Although this is a very simplistic representation of the underlying hidden states, we found it to be highly effective and efficient for the task of QRS detection.

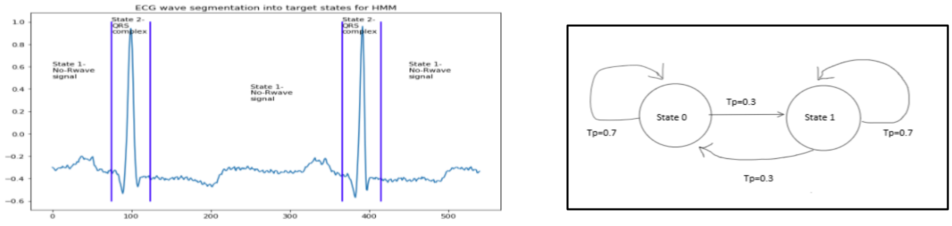


Figure 2: (a) Binary state model for HMM (b) Transition probabilities for HMM model

Next, we decided on the transition probabilities as shown in the figure 2 above for change from each state to the next state and also to itself. Intuitively, we want a state to continue until a definite state change event has occurred in which case, it should transition to the next state. We found that letting the HMM continue with 70% probability in the same state, especially state 0 minimized the chance of state transitions due to P-wave, while a 30% state transition probability was good enough to still change the state when a legitimate QRS event occurs.

6.2 Experiment Setup:

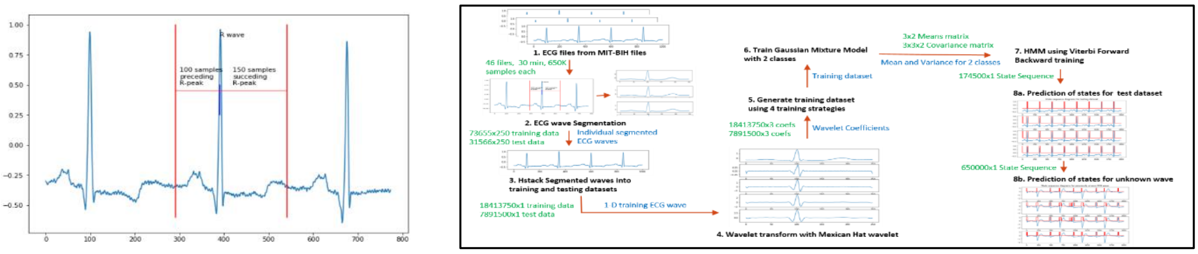
We first downloaded all the ECG records from MIT-BIH arrhythmia database locally using the WFDB [11] python package. Next, we defined the experiment pipeline as shown in the flowchart below. Each ECG signal is preprocessed using continuous wavelet transform which maintains the time component of the signal so essential for HMM. As recommended in [4], Mexican Hat wavelet is used, and we select only the 2,3 and 4th dyadic scales as the first scale is noisy.

We implemented four training strategies, detailed in the next section, to identify which one of them results in the best performance. As arrhythmias impact the temporal form as well as the amplitude of signals, we realized that the model that is trained on the signals of a patient can learn latent features of that patient's ECG very efficiently, however it fails to generalize when trying to predict unseen signals. To get a true sense of model performance, we evaluated the four models using two evaluation datasets which is detailed in section 6.4 below.

6.3 Training Strategies:

In the training phase, we explore four ways of creating training datasets. The training process followed for each training dataset is the same, the only change is in what data out of the pool is used to train the model.

ECG record segmentation to build training pool – As mentioned in section 3 above, the MIT-BIH dataset consists of 46 ECG waves, each being a 30-minute recording with 650K samples. To create the training dataset, only 45 recordings will be used reserving one recording for testing. We split each of the 45 recordings iteratively into its constituent ECG waves as follows - Using the annotation files that store the R-peak positions as an anchor point for each wave and using a window of size 250 centered at this anchor point, split the larger signal to pick 100 records immediately preceding the R-position and 150 records immediately succeeding the R-position. This process is performed for all 45 recordings for each annotated R-peak to create the pool of individual segmented ECG waves. Out of this pool, 70% are randomly selected and used to build the training dataset while the remaining 30% are used as testing dataset. This process ensures that model is exposed to latent characteristics of all patients, so it can generalize better. Because of the way we segmented, we know that the expected R-peak is at 100th position for each segmented wave.



1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
0
100
200
300
400
500
600
700
800
R wave
100 samples preceding R-peak
150 samples succeding R-peak
1. ECG files from MIT-BIH files
46 files, 30 min, 650K samples each
2. ECG wave Segmentation
73655x250 training data
31566x250 test data
Individual segmented ECG waves
3. Hstack Segmented waves into training and testing datasets
18413750x1 training data
7891500x1 test data
1 D training ECG wave
4. Wavelet transform with Mexican Hat wavelet
18413750x3 coefs
7891500x3 coefs
Wavelet Coefficients
5. Generate training dataset using 4 training strategies
Training dataset
6. Train Gaussian Mixture Model with 2 classes
3x2 Means matrix
3x3x2 Covariance matrix
Mean and Variance for 2 classes
7. HMM using Viterbi Forward Backward training
174500x1 State Sequence
8a. Prediction of states for test dataset
650000x1 State Sequence
8b. Prediction of states for unknown wave

Figure 3: (a) ECG wave segmentation

(b) Experiment Pipeline flowchart

Training pipeline - The training data selected is then passed through the wavelet transform and the 2,3 and 4th scales are picked up to generate the training dataset. This is fed to a Gaussian Mixture Model which identifies the mean and variances of best fitting gaussians for the two states. These parameters are then used in training the HMM using Viterbi forward-backward algorithm.

a. **Strategy 1 - Template ECG waveform** – The idea behind this strategy is that the ECG waveform follows the same sequence of states for every person every time. The proportion of time spent in each segment is also largely same. Given this prior knowledge about the ECG signal, we use a single ECG wave to train the HMM. This single ECG wave is of size 250 samples with its R-peak present at 100th sample.

b. Strategy 2 – Use the entire training pool – Here we try to feed the model as much variety of information as possible from the training pool. The segmented waves are concatenated to form a long ECG wave that acts as the training record for the model. The model parameters are learned on this single long ECG wave.

c. Strategy 3 – Training dataset + PCA – In this method, we follow the same steps as strategy 2 above to create a single training wave comprising of all segmented training waves. Before we train the model, we perform PCA decomposition on this to increase the variance between the 3 wavelet coefficients. It is then fed to GMM and HMM for training.

d. **Strategy 4 – Training dataset + ICA –** In this method, we follow same steps as strategy 3 but use ICA instead of PCA.

6.4 Evaluation metric calculation and testing strategies:

HMM being a state machine, it is important to interpret the basis of the evaluation metrics correctly. The definitions considered for True Positive, False Positive and False Negative are given below for clarity.

True Positive (TP) – For a point in time in the ECG wave, when the HMM state is 1 and the actual R-peak is present, it will be counted as TP.

False Positive (FP) - For a point in time in the ECG wave, when the HMM state is 1 but the actual R-peak is not in the vicinity of 10 samples on each side, it will be counted as FP.

False Negative (FN) - For a point in time in the ECG wave, when the HMM state is 0 and the actual R-peak is present, it will be counted as FN.

In addition to Precision and Recall, we will also be comparing the average state width for R-wave because a good model is expected to be in state 1 only for the duration of QRS complex.

Testing Strategy – As briefly described in section 6.3, we will use two datasets to evaluate the performance. The first dataset is the 30% data from the overall segmented ECG wave pool reserved for testing. The evaluation on this data will tell us how the model is performing on data it has already seen during training. The second dataset will be a recording that was not used to build the pool. Evaluating on this signal will test whether the model is able to generalize on unseen ECG features.

6.5 Results:

The below figure summarizes the metrics obtained for each of the four training strategies and two evaluation methods.

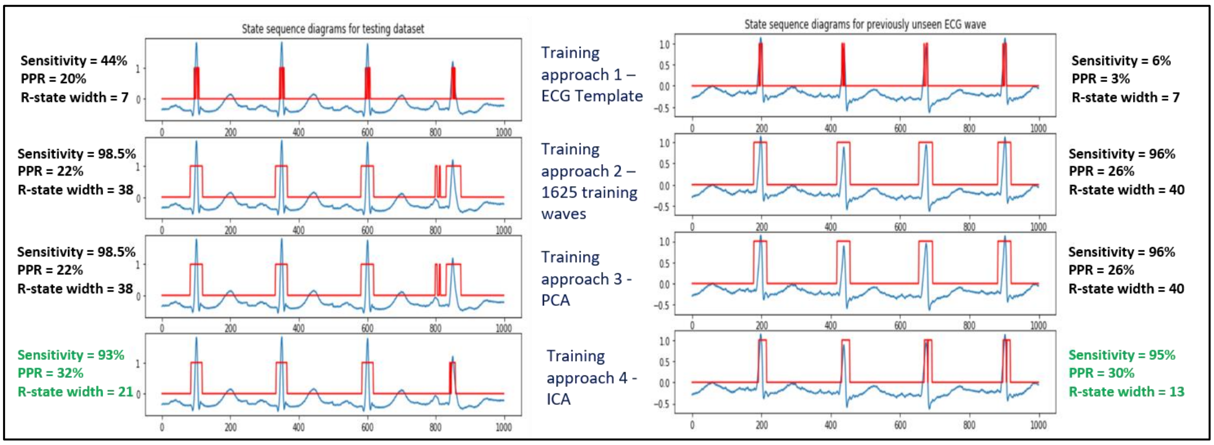


Figure 4. Experiment results showing metrics, sample state values with underlying ECG waves for 4 training and 2 testing strategies

7. Discussion:

7.1 CNN:

For the CNN model, we reached decent performance on the ECG of unseen patients, but strictly speaking, we can’t draw a conclusion that the CNN model generalizes well. To test generalization, we need to collect ECG recorded in different environments and different Signal-Noise Ratio (SNR) levels and see how the model performs on them, so simply testing model on the ECG of unseen patient in one dataset doesn’t provide solid conclusion on the generalization of CNN model.

Then focusing on the result the model produced, it’s clear that sensitivity is worse than positive predictivity, which means that false negatives (FN) are more than false positives (FP). This is due to the huge imbalance between R-peaks and non-R-peaks: among 650000 points in each record, there are at most 3000 R-peaks, thus, even if the model predicts all points as non-R-peak, it classifies more than 99.5% points correctly, and the cross-entropy loss function outputs low value in this case. Therefore, to make the output of cross-entropy small, the model has the tendency of classifying more points into non-R-peak, which results in more false negatives.

To improve the CNN model, we put forward three suggestions. First, changing objective function to generate more cost on false negatives, so the model is not inclined to classify more points into non-R-peak. Second, notice that the input of CNN model is a 301-d vector, which corresponds to 0.83 seconds long signal. After feeding into CNN, the feature of 0.83 seconds signal is extracted. We can let CNN capture the feature of short time signal as well by adding another branch of input that corresponds to 0.1 seconds long signal. Combining global and local level of feature, MLP can distinguish them better. Third, the CNN model is proposed based on ECG as graphical representation of heart activity rather than time series. However, the values in ECG are voltage values rather than RGB or grayscale, so raw ECG is intrinsically time series. Then to capture the repetitive pattern of time series, recurrent neural network (RNN) is a better option than CNN.

7.2 HMM:

For the HMM models, performing ICA on the training dataset results in a balanced outcome between Precision and Recall. The Precision suffers due to false positives because the model transits between states due to P-wave peaks, noise and model instability. Within the scope of this experiment, the model trained with ICA features generalized well on unseen ECG waveform as well as previously seen features of the testing dataset. The precision can further be improved by increasing the variance between the samples of two states. It was observed that for ICA, the parameters of the two gaussians that represent the two states had overlap causing spike in false positives. Other methods to improve the precision would involve redesigning the model using HSMM and EHMM, using advanced preprocessing techniques and non-linear dimensionality reduction algorithms.

8. Conclusion:

Although CNN and HMM approaches were not able to achieve state-of-the-art performance on MIT-BIH datasets, this project was able to explore innovative training and testing methods, implement a CNN+MLP based anomaly detection algorithm and design a two-state HMM model previously unexplored in available literature. Based on the outcomes discussed in section 5.3 and 6.5, the project can be deemed as successful as the intent was to apply the learning of the underlying course in Machine Learning to real life signal processing problems. Although each aspect of the pipeline can be further refined and improved upon in the future, this work, in itself, contributes to the existing pool of literature in terms of comparing two leading ML approaches to R-peak detection, the model configurations that work and the results of experiments undertaken in that regard.

9. References:

[1] C. K. Roopa and B. S. Harish, "A Survey on various Machine Learning Approaches for ECG Analysis," International Journal of Computer Applications, vol. 163, no. 9, 2017.

[2] B. Frénay, M. Verleysen and G. d. Lannoy, "Improving the transition modelling in hidden Markov models for ECG," in European Symposium on Artificial Neural Networks - Advances in Computational, Bruges (Belgium), 2009.

[3] R. V. Andreão, B. Dorizzi and J. Boudy, "ECG Signal Analysis Through Hidden Markov," IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, vol. 53, no. 8, pp. 1541-1548, 2006.

[4] R. V. ˜. Andre˜ao and J. ´. e. ˆ. Boudy, "Combining Wavelet Transform and HiddenMarkov," EURASIP Journal on Advances in Signal Processing, vol. 2007, 2006.

[5] N. P. Hughes, L. Tarassenko and S. J. Roberts, "Markov Models for Automated ECG Interval".

[6] D.A.Coast, M. R, G.G.Cano and S.A.Briller, "An approach to cardiac arrhythmia analysis using hidden Markov models," IEEE Transactions on Biomedical Engineering, vol. 39, no. 9, pp. 826-836, 1990.

[7] E. J. d. S. Luza, W. R. Schwartzb, Guillermo Cámara-Cháveza and c. David Menottia, "ECG-based heartbeat classification for arrhythmiadetection: A survey," Computer methods and programs in biomedicine, vol. 127, pp. 144-164, 2016.

[8] Y. Kaya, H. Pehlivan and M. E. Tenekeci, "Effective ECG beat classification using higher order statistic features and genetic feature selection," Biomedical Research, vol. 28, no. 17, pp. 7594-7603, 2017.

[9] R. Rodrígueza, A. Mexicano, J. Bila, S. Cervantes and R. Ponce, "Feature extraction of electrocardiogram signals by applying adaptive threshold and principal component analysis," Journal of Applied Research and Technology, vol. 13, pp. 261-269, 2015.

[10] L. G, G. R, W. F, W. K, O. A, N. H and a. Contributors, "PyWavelets - Wavelet Transforms in Python," 2006. [Online]. Available: https://github.com/PyWavelets/pywt. [Accessed 18 11 2018].

[11] C. Xie and J. Dubiel, "WFDB," [Online]. Available: https://wfdb.readthedocs.io/en/latest/index.html. [Accessed 01 10 2018].

[12] Zhu H, Dong J. An R-peak detection method based on peaks of shannon energy envelope. Biomed Signal Process Control.

2013;8:466–74.

[13] Chalapathy, Raghavendra. “Anomaly Detection Using One-Class Neural Networks.” 18 Feb. 2018, arxiv.org/pdf/1802.06360.pdf.

[14] Trahanias PE. An approach to QRS complex detection using mathematical morphology. IEEE Trans Biomed Eng.1993;40:201–5.

[15] Xiang, Yande. “Automatic QRS Complex Detection Using Two-Level Convolutional Neural Network.” BioMedical Engineering Online, Jan. 2018, doi.org/10.1186/s12938-018-0441-4.

[16] MIT-BIH Arrhythmia database available online at https://physionet.org/physiobank/database/mitdb/