CS 2756 Data Mining Project Report

An electrocardiogram (ECG) signals classification by recognizing the arrhythmia.

Yongjie Yang, Xueteng Qian

4/23/2022

Catalog：

Abstract:

Heart disease has been a problem for decades. Although many modern medical devices can provide us with very portable measurements, the results are often unsatisfactory. One hand of this is due to the immaturity of the hardware, and another hand is due to the incompleteness of four software algorithms. This report will describe the later processing of cardiac signals and obtain an excellent software algorithm result through data analysis on a medical basis. In general, we did two different learning models to train the computer (even deployed on mobile devices) to be good at detecting arrhythmia (a problem with the rate or rhythm of the heartbeat). The dataset we are trained and tested is the MIT-BIH arrhythmia database, collected from the real world. The results of two different methods are compared in this report. The two models are support vector machines (SVM) and convolutional neural networks (CNN). When tested on 48 ECG records, the SVM and CNN detection approaches performed acceptable results.

Introduction and Motivation:

There is a scary number about heart disease; about 659,000 people in the United States die each year from various heart diseases. The most dangerous part, in other words, is that nearly every 36 seconds, someone dies of heart disease. So, when it comes to heart disease, scientists and doctors have been actively and relentlessly trying to reduce mortality and even have developed to analyze ECG signals using mobile devices. Considering these trends, monitoring heart activity is suitable for human development. It is also essential to compute and analyze the heart signal at the back-end.

The electrocardiogram (ECG) signal reflects the heart's electrical activity observed from strategic points of the human body and is represented by quasi-periodic voltage signals. ECG signals are measured using sophisticated instruments with various leads in a hospital setting. However, because our database is relatively simple, the recordings were directly used from the two-lead structure. Hence, there are two signals; the upper signal is a modified limb lead II (MLII), obtained by placing the electrodes on the chest. The lower sign is usually a modified lead. As figure 1 shows, we only focus on five peaks in ECG signals, which are denoted by the letters P, Q, R, S, and T waves. Then the diagnosis of the signal relies on the morphology of the waves, as well as the duration of each peak and the segments that make it up. In a simple sentence,

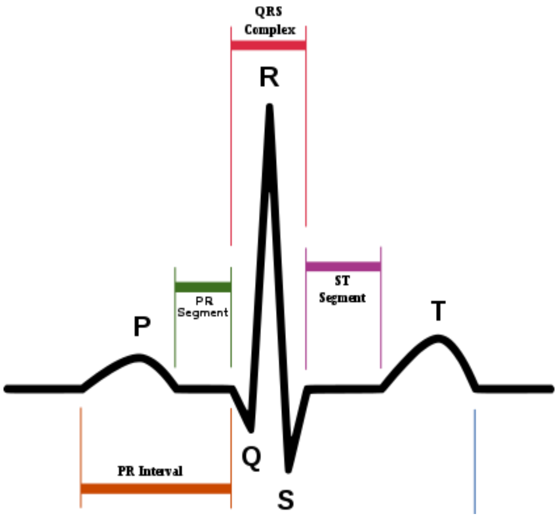

Figure 1: Electrocardiogram Signal

if the amplitude of each wave peak or the spacing between each waveband is very weird, we can detect and judge it as an abnormal heart signal. Therefore, monitoring the ECG segment of the movement is vital for doctors to treat the disease and gives researchers like us the medical basis to screen and diagnose heart diseases scientifically.

Based on medical judgment, we have classified five different heart diseases. Table 1 below shows that these five heartbeats are commonly and highly vulnerable for patients. N represents regular beats, and L and R respectively represent left and right bundle branch blocks. A means premature atrial contraction, and V symbolizes ventricular premature contraction. Figure 1 shows that the medical theory instructs that the atrial premature condition will have a super longer P peak wave. The left bundle branch block will have a broader and larger Q peak wave. The right bundle branch block will have a broader and deeper S wave. And the last, ventricular premature, represents the QRS complex interval has a longer wavelength.

(N)	Normal beats
(L)	Left bundle branch block
(R)	Right bundle branch block
(A)	Atrial premature contraction
(V)	Ventricular premature contraction

Table 1: Five Heart Diseases

This report provides a novel method for classifying ECG signals, using a convolution neural network to identify each heart disease. Overall, our pre-processing signal method is based on wavelet transform algorithms. In other words, use the wavelet transform to capture the denoised and segment heart signal and then separate the signal periodically. Subsequently, SVM and CNN are used to carry out the same training model and then compared with the test set. Finally, we tested both classification algorithms with signals acquired in real-time from the MIT-BIH database. Both test results methods were higher than 95 percent.

Database (©*https://archive.physionet.org/physiobank/database/html/mitdbdir/intro.htm*):

The source of the ECGs included in the MIT-BIH Arrhythmia Database is a set of over 4000 long-term Holter recordings obtained by the Beth Israel Hospital Arrhythmia Laboratory between 1975 and 1979. Approximately 60% of these recordings were obtained from inpatients. The subjects were 25 men aged 32 to 89 years and 22 women aged 23 to 89 years because infants and adolescents have meager death rates from heart diseases.

The database contains 23 records (numbered from 100 to 124 inclusive with some numbers missing) chosen at random from this set, and 25 records (numbered from 200 to 234 inclusive, again with some numerals missing) selected from the same group to include a variety of rare but clinically essential phenomena that would not be well-represented by a small random sample of Holter recordings. Each of the 48 records is slightly over 30 minutes long. Therefore, this amount of data is enormous and can meet our requirements.

There are two groups of recordings from this database:

1. The first group is intended to serve as a representative sample of the variety of waveforms and artifacts that an arrhythmia detector might encounter in routine clinical use. A table of random numbers was used to select tapes and then to select half-hour segments. Segments chosen in this way were excluded only if neither of the two ECG signals was of adequate quality for analysis by human experts.

2. Records in the second group included complex ventricular, junctional, and supraventricular arrhythmias and conduction abnormalities. Several of these records were selected because features of the rhythm, QRS morphology variation, or signal quality may be expected to present significant difficulty to arrhythmia detectors; these records have gained considerable notoriety among database users.

Signal Pre-processing:

Wavelet transforms

There is an essential difference between the ordinary data recorded on paper and our dataset. Our data is based on a discrete sequential signal. Since an average human heart rate ranges from 60 to 80 beats per minute, though it is a continuous but irregular signal, there is no definitive number of heartbeats per minute. Hence, we need to recognize every heartbeat signal per minute. In addition, a body is not only the sound of the heartbeat, but we also have all kinds of other body tissue noise, such as the noise of breathing in the lungs near the heart. Therefore, based on these non-negligible reasons, we adopt the signal processing of wavelet transforms. The basic idea is to compute how much of a wavelet is in a signal for a particular scale and location, which is the convolution computation. In other words, a signal is convolved with a set of wavelets at various scales. Specifically, we first used a wavelet to find the R peak of each heartbeat. Then, we get transformed 360 samples for each heartbeat for a whole heartbeat period. Before R peak, we took 160 points; after R peak, we took 200 points as the data flow to judge abnormal signals later. There are 48 recordings of heat signals, and each of them could be 30 minutes, so there are around 22,000 samples in total. The whole process of wavelet transformation we used in this project calls the python packages directly.

After obtaining the data denoised by wavelet transform, we proceed with a simple data selection process. To get more accurate signals, we eliminated the first five and last ten samples of each heartbeat. Although this may remove the evident P- and T-associated signs from a complete call, it can significantly increase the accuracy of a cardiac signal; after all, we don’t need t-waves to distinguish our five heart diseases.

Model:

Since our database is labeled with different heart diseases, we are inclined to use supervised learning models. Hence, we used a traditional model that we've seen in our class, and then we used an advanced machine learning model for data mining. One model is a support vector machine, and the other is called the convolutional neural network.

Support Vector Machine (SVM)

The first model we used is our traditional way, a support vector machine, which we have learned in the class. In our project, SVM is a supervised learning model with associated learning algorithms that analyze data for classification and regression analysis. As figure 2 shows, the basic principle of SVM is just using binary classification to manage the data. SVM maps training samples to points in space to maximize the width of the gap between the two categories, and overfitting is handled by maximizing the margin of the decision boundary.

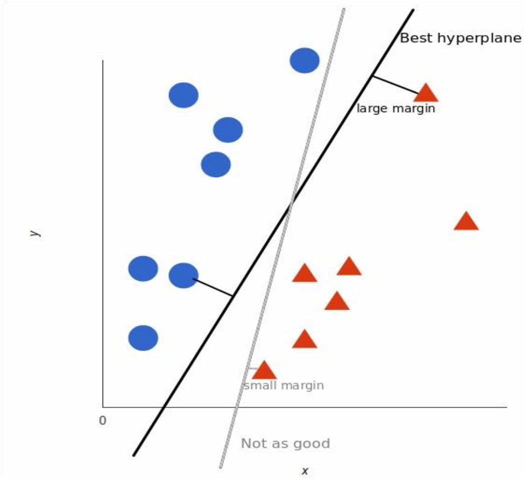

Figure 2: Support Vector Machine

There are many advantages to choosing the SVM; for our project, it indicates that it is robust to noise. Besides, SVM can handle irrelevant and redundant datasets. In our model, we used 30 percent of the data as the test set and the rest as the training set to train our SVM model.

Convolutional Neural Networks (CNN)

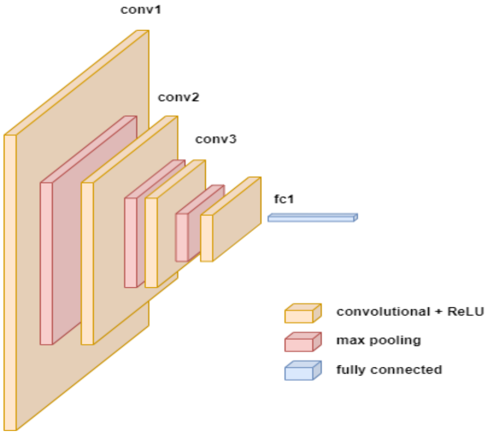

Figure 3: Network Structure

Deep neural learning networks are recently viral and advanced, and we used convolution neural networks in this project.

Our network architecture comprises convolutional and pooling layers for this model. The convolutional layers are 1-D convolution on the preprocessed ECG data points, and the activation function of this layer is ReLU. We concatenated the max-pooling layer to the convolutional layer to form a block. The network comprises three blocks followed by a fully connected layer to map the network output to the five classes. The reason that we are using three blocks for this network is that this has the best performance. We have tried to use two blocks and three blocks, but fewer blocks will underfit the dataset, and more blocks will overfit the dataset, and we find out that three blocks will produce the best result. The loss function we use is cross-entropy.

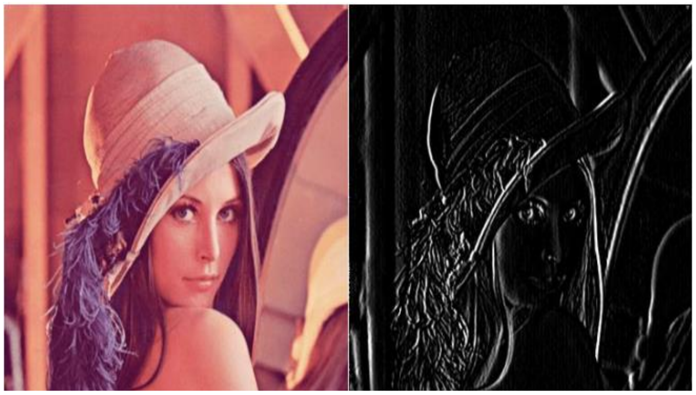

Figure 4: An example of the convolutional layer

The essential operation of the convolutional layer is to convolute the input with the kernel using a window that has the same size as the kernel. We use a convolutional layer because convolution could help extract the essential features of the ECG signal. We could use a picture as figure 4 to visually demonstrate this; we can see that the picture on the right is after convolution processing, and the edges of this picture are being extracted. With the updating weights, the network could learn by itself to remove critical information for classification.

We also used a pooling layer. The process of the pooling layer is simple: find the maximum value of the window as its output and keep moving the window. If we do this process on the picture, we will find that the image will become a blur. You might wonder why do we want to make our features blur? Is not that harmful? There are mainly two reasons for using the pooling layer. The first one is that this layer could reduce the parameters for further calculations. Another reason is that blurring will help us resist noise. Intuitively, on a picture, the more precise the image is, the more edges and details we get, but sometimes, we do not need that much detail. For example, we do not need to use the texture detail of the hat’s decoration to classify the person. Thus, using a pooling layer is helpful for our classification.

Applying to our project, the whole flow process looks like the flow chart below. We implemented the convolutional neural network using TensorFlow. We used 30 training epochs, 0.001 learning rate with Adam optimizer, and 256 batch sizes for the training parameters. We just used the default values because we do not want to fine-tune these parameters to fit the dataset so well that the meaning of comparison with SVM might be lost. Besides, we still use 30 percent of the data as a test set for the controlled trial's purpose.

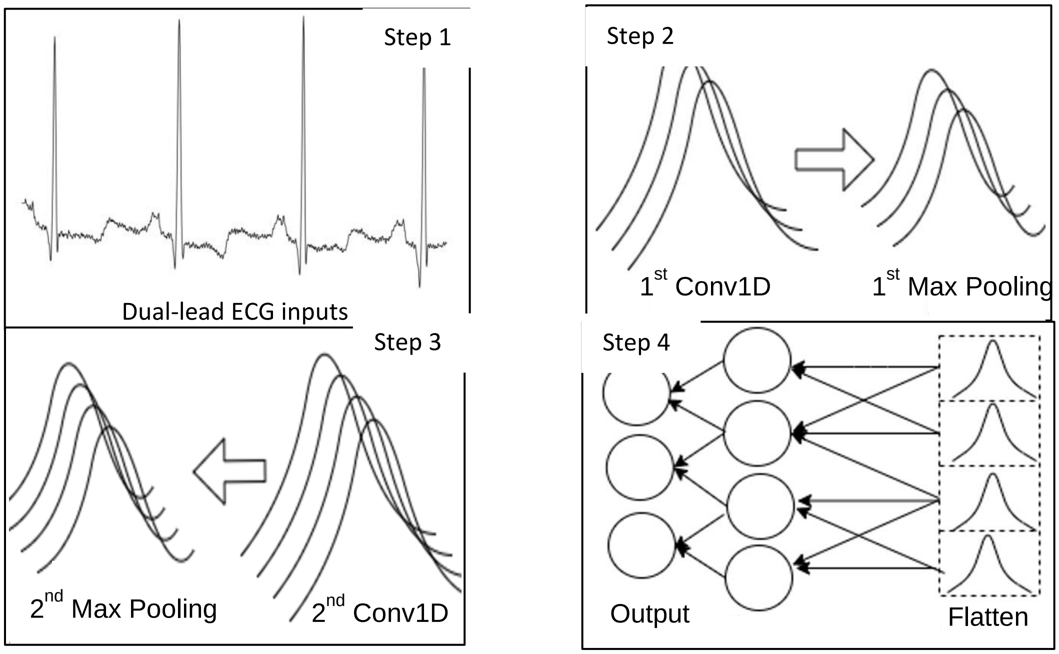

Figure 5: Flowchart of our CNN model

Evaluation:

Our evaluation method yielded good results by using the F-1 score calculation. Both models of their performance are higher than 95 percent. The total testing accuracy of the SVM is around 0.988, and CNN's accuracy is about 0.992. Then next, Tables 2 and 3 show the performance of both models for each heart disease. Overall, the results of CNN were slightly better than those of SVM but significantly higher in premature atrial contraction. To deal with this kind of high-precision picture of signals, CNN is better than SVM. We will discuss more details of data analysis in the 'Results' and 'Discussion' parts.

SVM	**N**	**L**	**R**	**A**	**V**
Precision	0.999	0.977	0.978	0.721	0.955
Recall	0.979	0.993	0.995	0.985	0.993
F-1 score	0.989	0.985	0.986	0.832	0.974

Table 2: Evaluation results of SVM

CNN	N	L	R	A	V
Precision	0.998	0.996	0.995	0.898	0.979
Recall	0.995	0.996	0.998	0.944	0.989
F-1 score	0.996	0.996	0.996	0.92	0.984

Table 3: Evaluation results of CNN

Results:

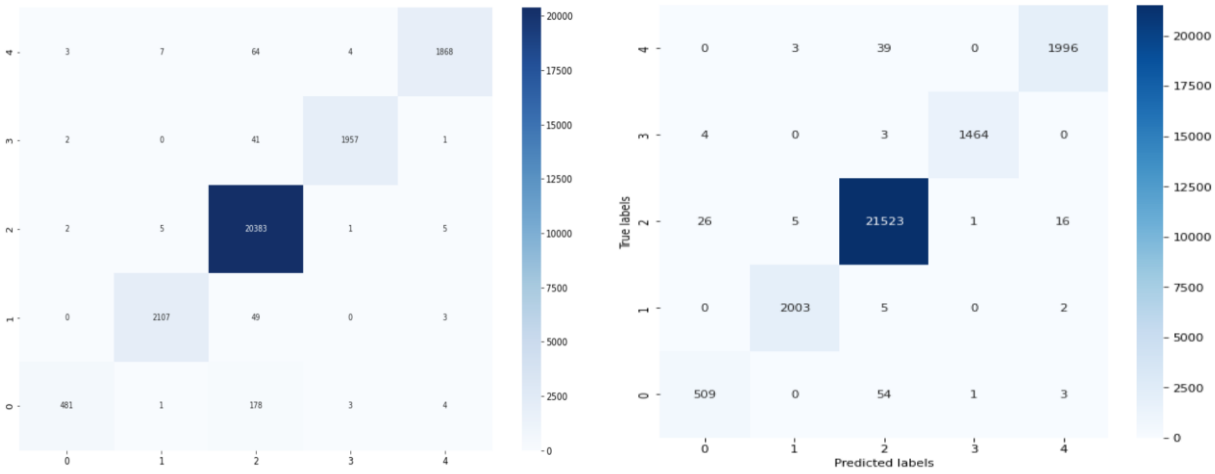

Figure 6: SVM (left) and CNN (right) results

Figure 6 above shows our experimental results using two different training models, SVM and CNN. This part mainly elaborated on the two kinds of experimental results that have an excellent comparative effect on both models.

We sampled about 27,000 data points, and we can see that these data are distributed in the grid above. In the coordinate system, our ordinate represents accurate data labels. After the signal processing stage, these are defined directly by the original dataset from the MIT-BIH database. Comparatively speaking, the abscissa represents the data labels generated by our training models. And we have replaced the tedious list of heart diseases names with numbers. 0 illustrates the premature atrial contraction. One means the left bundle branch block. And 2 means regular heart activity. 3 means right bundle branch block, and the last one, 4, represents ventricular premature contraction.

The correct date should be distributed on the diagonal as the results turned out. And we can see that most of the data focused on the average heartbeat, which was pretty much what we expected.

The fundamental difference between the two models is not too distant. We know that the SCM and CNN perform with comparable accuracy against the same dataset if given equivalent training. However, there is a minor difference in the premature atrial contraction between the two models. The SVM has the more significant error in classifying this heart disease as harmful data. Negative numbers are inevitable. Computers cannot process 100 percent of our heartbeat. But if we train and test more recordings, the data mining will be more profound. Besides, if we apply the cross-validation method instead, in other words, if given as much training and computational power as possible, CNN tends to outperform SVM.

Discussion:

We have more practical problems to solve in these two application models, mainly from the following three aspects to carry out our discussion part. And we hope to have the opportunity to optimize SVM and CNN models in the following work.

1. *Data pre-processing:* The biggest problem is our dataset; our dataset adoption from the MIT-BIH database is unbalanced because the patient's heart frequency was only a tiny part of the original dataset; most samples were collected from individuals' normal heart activities. Therefore, this situation leads to the imbalance of our training model. From the diagram in the last part, we can see that most of the samples are concentrated in the average heart rate part, which is more than a multiple of an order of magnitude than other heart diseases, so it may be a potential cause of the overfitting. The most intuitive method is re-sampling to solve this bias large data offset. Redistributing the sample size of the five diseases will significantly increase the diversity of the model and make the models we trained to have clinically valuable.

2. *Model training:* Because of the computational capacity of our laptops, we didn't implement very concrete training models. This leads to potential problems with our training models, such as overfitting or underfitting. We can apply the K-fold cross-validation to improve the training sense, at least ten folds. We can divide our original samples 10-fold, and each subset of data is used as a validation set, and the remaining 9 (K-1) subset data is used as a training set. In this way, 10 (K) models can be obtained. The 10 (K) models are evaluated in the validation set, respectively, and the mean squared error (MSE) is added and averaged to obtain the cross-validation error. Then the cross-validation effectively utilizes limited data. The evaluation results could be as close as possible to the model's performance on the test set, which can be used as an indicator of model optimization.

3. *Model evaluation:* In complex machine learning or data mining models, we must constantly verify the characteristics, such as feasibility, authenticity, and accuracy. Hence, we can use different evaluation methods to re-assess our models, for example, the receiver operating characteristic curve (ROC curve). ROC curve combines the actual case and a false positive case rate by graphical method, accurately reflecting the relationship between the real case rate and false positive case rate of a certain leaner and is a comprehensive representation of detection accuracy. ROC curve has no fixed threshold, allowing the existence of intermediate states, which is conducive to users, such as doctors, to weigh the influence of missed diagnosis and misdiagnosis

by combining professional medical knowledge and selecting a more threshold as a diagnostic reference value. If we could implement a ROC curve to determine whether the model is overfitting or underfitting, it would be good to adjust the model training process by trying different epochs. In addition, a medical training model is often not as simple as the one in our project because we need a more accurate and complete model to avoid doctors' misdiagnosis. Hence, we must consider more aspects of our model evaluation. Specifically, we can implement sensitivity and specificity by applying the methods of positive predictive value (PPV) and negative predictive value (NPV) to finalize a concrete model analysis.

Contribution:

As for the division of labor for this project, we both contributed equally to this one. We mapped out the whole project together. First, for the signal pre-processing part, we jointly participated in and wrote the code of the wavelet transform and achieved its fundamental idea. In addition, we are also responsible for the preparation equally for the final summary work, such as the presentation slides and the report.

Yongjie Yang - From starting to learn about ECG signals and understanding the process of denoising the signals, I have learned a lot. I mainly implemented the work of the support vector machine. For SVM, I got an acceptable result from the customer design parameters of the training model, and then the results were verified against the test set. And for this project, I also learned a lot about the knowledge of data mining; for example, we not only used the traditional way to do the project, and we also used more advanced algorithms to classify the signals. And finally, we compared the performance of these two methods.

Xueteng Qian – I was mainly responsible for designing the whole process of convolutional neural networks. As I mentioned in the Method section, it starts from the input of data images from the signal processing end to the design of the various layers, such as the convolution and pooling layers. In conclusion, I have learned a lot about data mining knowledge from this project, for instance, data processing and analysis.

Conclusion:

In conclusion, this project let us experience the process of doing data mining in real life. Not only do we learn a lot of data about daily life from our classes, but I also think it is essential as medical evidence. The project let the knowledge learned could be closely related to medical development so that our living standards will be significantly improved. From a theoretical point of view, our first exposure to neural network learning has dramatically improved our research background, and we have mastered various traditional data mining methods.

Reference:

The overall code structure is modified according to the existing code on GitHub.

Our project GitHub link: https://github.com/lijinlunbeng/CS-2756-FINAL-PROJECT

The reference website links:

1. https://github.com/mondejar/ecg-classification
2. https://github.com/physhik/ecg-mit-bih
3. https://github.com/Vidhiwar/multimodule-ecg-classification
4. https://www.nature.com/articles/s41598-021-84374-8
5. https://www.csdn.net/tags/MtjaIg1sODM1NDktYmxvZwO0O0OO0O0O.html
6. https://towardsdatascience.com/the-wavelet-transform-e9cfa85d7b34
7. https://archive.physionet.org/physiobank/database/html/mitdbdir/intro.htm