Neurocomputing 417 (2020) 64–73

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Knowledge-shot learning: An interpretable deep model for classifying imbalanced electrocardiography data

Yen-hsiu Chou [a,b], Shenda Hong [c], Yuxi Zhou [a,b], Junyuan Shang [a,b], Moxian Song [a,b], Hongyan Li [a,b,*]

[a] School of Electronics Engineering and Computer Science, Peking University, Beijing, China
[b] Key Laboratory of Machine Perception (Ministry of Education), Peking University, Beijing, China
[c] Department of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, USA

ARTICLE INFO

Article history:
Received 4 October 2019
Revised 16 March 2020
Accepted 17 July 2020
Available online 28 July 2020
Communicated by Zhu Yu

Keywords:
Heartbeats classification
Imbalance problem
Neural network
Interpretable Model

ABSTRACT

Classification of Electrocardiogram (ECG) data has been an important research topic in the machine learning area for many years. Recently, deep learning methods have been used in classifying ECG data and have achieved superior results than traditional methods. However, in real-world applications, two challenges existing deep learning methods cannot handle well – imbalanced data and model interpretability. In this paper, we propose an ECG classification method named Knowledge-Shot Learning (KSL) that can handle with the above two challenges. KSL constructs a novel neural network architecture that can be effectively trained on imbalanced ECG data. Besides, KSL can also extract interpretable feature vectors and give support cases as result explanations. Moreover, KSL can even classify unseen diseases if provided with the necessary medical knowledge. Experiments on real-world ECG data show that KSL improves 10.00% of F_1-score on imbalanced classes, and 43.75% of F_1-score on unseen classes, compared with the second-best baseline. KSL also provides interpretable results that are consistent with medical domain knowledge.

1. Introduction

Heart disease is the leading cause of death in the world. For example, ventricular fibrillation, one of the most likely mechanisms for sudden death, is leading the cause of fatality among patients with coronary heart disease [1]. Accurately and timely treatment is crucial – the survival rate of patients receiving rapid defibrillation is 15.3% higher than the defibrillation time by 2 min or more [2].

With the popularity of wearable devices, it is now much more convenient to collect patients' ECG data, which records physiological activities of the heart over a while. Thus, the abundant heart-related health information provides an opportunity for machine learning based automated ECG diagnosis tools to provide a more accurate diagnosis for patients to get medical treatment for inspection or first aid. We classify these existing methods as three categories, namely, feature engineering methods [3–7], deep learning methods [8–22] and combined methods [23–26]. Deep learning methods have surpassed the traditional feature extraction method in ECG diagnosis, due to its ability to automatically extract effective features and classification using Convolution Neural Network (CNN) [10,11,23,18], Recurrent Neural Network (RNN) [12–15] or CNN + RNN [22,16,17]. Moreover, combined methods further improve the performance by combining the handcrafted domain feature with deep learning methods (more details see Section 2).

However, despite the progress the aforementioned methods have made, there are still several challenges that they ignore or can not handle well as follows:

- **Imbalanced Data**: In the real world, most people are healthy; some people have some common heart diseases, only a small fraction of people have rare diseases. This situation leads to a serious imbalance problem in almost all collected ECG data. However, most ECG classification methods are easily biased to the number of categories with a large number of samples, resulting in poor performance of category classifications with fewer samples. This problem makes it harder for deep learning methods to achieve good performance on these rare diseases. However, it is much more important to classify rare diseases correctly. For example, ventricular escape happens rarely, but

* Corresponding author at: School of Electronics Engineering and Computer Science, Peking University, Beijing, China.
E-mail address: leehy@pku.edu.cn (H. Li).

https://doi.org/10.1016/j.neucom.2020.07.045
0925-2312/

it is hazardous that it may lead to death in a few minutes, commonly due to heart attack or medication side effect [27]. For imbalance problems, resampling is usually used to solve this problem, but if the sample size is too small, the category will be overfitted and the features of the category will still not be learned.

- **Classifying Unseen Categories**: As an extreme case of imbalanced data, we consider how to classifying classes that have never happened in existing training data, which is also likely to happen in a real-world application. However, most existing deep learning methods were designed to learn from ECG data, but not ECG medical domain knowledge. It is hard to equip them with medical domain knowledge for classifying unseen categories. For the classification of invisible categories, the general strategy is to correspond the extracted part of model features to the knowledge domain that humans can understand, and then artificially give the features of the invisible categories. However, doing so will cause the classifier to only use artificially given features to classify, which will cause the classification performance of other visible categories to decrease.
- **Model Interpretability**: Aside from classification results, interpretable explanations that can support the results are even more important for cardiologists to make a clinical decision carefully. Classification results solely without any medical evidence support are absolutely not acceptable for patients [28,29]. However, most existing deep learning methods are too sophisticated to be understood directly by a human. In fact, they are often regarded as black-box models without necessary interpretability. These methods can only tell "what is it" but not "why is it". Most interpretable methods are simpler and limited in performance. Therefore, it is difficult to build an interpretable high-performance model.

To solve the above problems in deep learning methods for ECG classification. In this paper, we propose an interpretable deep model named `KSL` to classifying imbalanced ECG data. In detail, `KSL` first builds a knowledge explainer neural network to extract knowledge vectors, which are feature vectors that in accord with engineered features and golden rules from medical domain knowledge. Then, `KSL` build knowledge evaluator neural network to measure the pairwise similarity between knowledge vectors. The ground-truth of pairwise similarity would be close to 1 if paired knowledge vectors (also the paired ECG data) are more likely from the same class; otherwise it would close to 0. Finally, when classifying a new ECG data, one can pick some candidate ECG data from the whole dataset, compute the similarity scores between the new ECG data and them via `KSL`, then vote to get the final results. Meanwhile, one can also use the candidate ECG data as interpretable medical evidential supports. Intuitively, we named our method "Knowledge Shot" because we classify a new labeled ECG data based on explained knowledge vector instead of ECG data directly (Knowledge), and we also use a pairwise similarity comparison instead of probabilities (Shot).

Our contributions are listed as follows:

- We propose a novel deep learning method `KSL` that can be effectively trained on imbalanced ECG data. As an extreme case of imbalanced data, `KSL` can even classify unseen diseases if provided with necessary medical knowledge.
- `KSL` provides easily understandable interpretable knowledge vector and evidential supports along with classification results. Thus it can be better used in the medical area.
- Experiments on real-world ECG data show that `KSL` improves 10.00% of F_1-score on imbalanced classes, and 43.75% of F_1-score on unseen classes, compared with the second-best baseline.

2. Related work

Existing ECG classification methods can be divided into three categories: feature engineering methods, deep learning methods, and combine methods. The three types of methods are introduced below. At the end of the section, we compare the similarities and differences between `KSL` and metric learning.

2.1. Feature engineering methods

In the past decades, with the help of computing technology, many algorithms have been proposed to classify ECG automatically. Most of these works rely on designing effective features based on medical domain knowledge and then utilize traditional machine learning methods to classification.

These effective features based on medical domain knowledge are easy to be understood by a human, such as whether P waves disappear or the QRS duration. For example, H Pürerfellner et al. [3] use the feature of whether P waves are detected to improve the performance of AF detection successfully. Carrara et al. [4] use linear and dynamic measurements of RR interval time series to extract feature and classify by logistic regression, k-Nearest Neighbors, and random forests. Many of these methods have been thoroughly investigated in previous literature [5,3,4,6,7]. While such traditional methods are interpretable, the process of presenting effective features is exhaustive, but these methods are limited in performance, and there is no way to detect ECG data for categories that have not been seen.

2.2. Deep learning methods

Recently, the deep learning method has surpassed the traditional feature extraction method in ECG classification, due to its ability to extract effective features and classification automatically [9]. These method using Convolution Neural Network (CNN) [10,11,23,18], Recurrent Neural Network (RNN) [12–15] or CNN + RNN [22,16,17]. For example, Xiong et al.[10], proposed a method for ECG classification using repeated 16 1-D convolutions with skip connections. Hannun et al. [11], proposed a method for ECG classification using 16 residual blocks with two convolutional layers per block. The convolutional layers have a filter width of 16 and 32*2m filters, where m is a hyper-parameter which starts at 0 and is incremented by 1 every fourth residual block. Schwab et al. [13] use RNNs with 1–5 recurrent layers that consist of either Gated Recurrent Units (GRU) or Bidirectional Long Short-Term Memory (BLSTM) units for ECG classification.

However, most deep learning methods equally treat every training samples, so that they can not be effectively trained on imbalanced ECG data, and there is no way to deal with this category without training samples. Besides, they are too sophisticated to be understood that is not applicable in medical situations.

2.3. Combined methods

In order to achieve higher performance, some research attempts to combine feature engineering methods and deep learning methods for ECG classification. Hong et al. [23,24] first explore expert features from the statistical area, signal processing area, and the medical area. Then, they build a deep learning neural network to extract features automatically. Finally, they combined these features and put them into ensemble classifiers. Shashikumar et al. [25] apply continuous wavelet transform to PPG data and train convolutional neural networks (CNN) on the derived spectrogram to detect AF. The combination of CNN output and features based on beat-to-beat variability and signal quality calculations provides significant accuracy improvements. Golrizkhatami et al. [26] proposed system utilizes a novel decision-level fusion of features for ECG classification by three different approaches: using the wavelet transform based morphological features representing localized signal behavior, statistical features exhibiting overall variational characteristics of the signal, and temporal features representing the signal's behavior on the time axis extracted by deep neuronetwork.

Although combined methods have shown better performance than individual methods, effective training is still not possible on imbalanced ECG data. Besides, the combined method cannot handle the case where there is no training sample for this category. Moreover, combined methods are also more complicated, so most of them are difficult to interpret.

2.4. Difference with metric learning

In the field of person re-identification, there are similar studies, metric learning, that extract features from input images and then calculate the similarity to solve the problem of the replacement of training samples in some categories [30–32], but the research in KSL and the field of person re-identification has the following differences: (1) There are fewer categories of heart disease than person re-identification, so there is no problem with the network being too large. Therefore, KSL design is to input the samples of each class into the neural network at the same time and select the most similar sample class as the output instead of just inputting two samples to determine whether they are the same class; (2) Since KSL can convert ECG to the knowledge vector corresponding to the golden role in medicine, KSL can artificially give the knowledge vector through the medical golden rule to determine the heart disease without ECG data. This enables KSL to classify unseen classes.

3. Our method

In this section, we introduce our method in detail. We first formulate the ECG classification problem and give the framework our KSL in Section 3.1. Then, we introduce two main components in KSL, namely, the knowledge explainer in Section 3.2 and the knowledge evaluator in Section 3.3. Finally, we show how to train KSL in Section 3.4.

3.1. Formulation

Formally, we use $\mathbf{X} \triangleq \{\mathbf{x}^{(1)} \ldots \mathbf{x}^{(n)}\} \in \mathbb{R}^{n\times t}$ denotes ECG data with n samples, each sample $\mathbf{x}^{(i)} \in \mathbb{R}^t$ has t data points and $\mathbf{Y} \triangleq \{y^{(1)} \ldots y^{(n)}\}$ denote classification labels, each label $y^{(i)} \in \{1 \ldots k\}$. So there are c diseases in these n ECG data, and use $\mathbf{x_c}$ to denotes an ECG data in c class.

Besides, we also denote $\mathbf{A} \triangleq \{\mathbf{a}^{(1)} \ldots \mathbf{a}^{(n)}\} \in \mathbb{R}^{n\times d}$ denotes n extracted knowledge vectors, each knowledge vector $\mathbf{a}^{(i)} \in \mathbb{R}^d$ has d attributes, and similarly $\mathbf{a_c}$ is used to denotes a knowledge vector in c class. Moreover, we use $\mathcal{H}$ and $\mathcal{G}$ denote knowledge explainer and knowledge evaluator introduced in Section 3.2 and Section 3.3 respectively. Notations is summarized in Table 1.

We cast the ECG diagnosis task into a multi-class prediction problem. To predict the unknown ECGs, KSL first build knowledge explainer neural network $\mathcal{H}$ to extract knowledge vectors $\mathbf{a}$ from ECGs. Then, KSL build knowledge evaluator neural network $\mathcal{G}$ to compute the pairwise similarity between the knowledge vectors $\mathbf{a}$. The overall framework of KSL is shown in Fig. 1.

Table 1
Notations.

Notation	Definition
$\mathbf{x} \in \mathbb{R}^t$	ECG data in t dimension
$y \in \{1, \ldots, k\}$	Label with k classes
$\mathbf{a} \in \mathbb{R}^d$	Knowledge vector in d dimension
$\mathbf{d} \in \mathbb{R}^l$	Difference vector in l dimension
$\mathcal{F}_\theta$	Feature extractor with parameter θ
$\mathcal{S}_\phi$	Pairwise similarity calculator with parameter ϕ
$\mathcal{D}_\varphi$	Pairwise difference calculator with parameter φ
$\mathcal{H}$	Knowledge Explainer
$\mathcal{G}$	Knowledge Evaluator
$\mathcal{D}_{xy} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$	Data set with ECG data and labels
$\mathcal{D}_{xa} = \{(\mathbf{x}^{(i)}, \mathbf{a}^{(i)})\}_{i=1}^n$	Data set with ECG data and knowledge vector
$\mathcal{D}_{ay} = \{(\mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$	Data set with knowledge vector and labels

3.2. Knowledge explainer

When doctors diagnose ECGs, they first observe the features of these ECGs (e.g. P wave amplitude, PR interval, etc.), and then use these features to classify ECGs. The function of knowledge explainer is similar to the observation of ECGs when doctors classify ECGs. Knowledge explainer extracts features from ECGs and treats these features as attributes of ECGs. We combine all attributes into a knowledge vector $\mathbf{a}$.

The way our method extracts knowledge vector from an ECG is using neural network to extract meaningful ECG features (e.g. P wave amplitude, PR interval, etc.) corresponding to medical domain knowledge as Eq. 1

$$\mathcal{F}_\theta : \mathbb{R}^t \to \mathbb{R}^d \tag{1}$$

The the knowledge vector $\mathbf{a} = \mathcal{F}_\theta(\mathbf{x})$ which contains numeric values that describe those features of ECG data.

We first randomly initialize the parameters of feature extractor θ, and then optimize it by gradient descent as follows,

$$\theta =: \theta - \alpha \nabla \mathcal{L}_{\mathcal{D}_{xa}}(\theta) \tag{2}$$

$$\mathcal{L}_{\mathcal{D}_{xa}}(\theta) = \frac{1}{\mid \mathcal{D}_{xa} \mid} \sum_{(\boldsymbol{x},\mathbf{a}) \in \mathcal{D}_{xa}} l(F_\theta(\boldsymbol{x}), \mathbf{a}) \tag{3}$$

where l is the empirical loss, which can be chosen as mean square error loss and α the learning rate.

For ECG of rare heart diseases, medical knowledge can be used to assist diagnose. However, in the general computer-aided ECG diagnosis methods, there is no way to judge the electrocardiogram of a rare heart disease that has not been seen. For the unseen class of heart disease, we use medical knowledge to obtain knowledge vector by medical domain experts. For example, whether ECG has wide and large QRS waves, irregular RR intervals, or P-waves missing, etc. These meaningful ECG patterns describe the characteristics of ECG data, such as morphological features, statistical features.

To map the knowledge vectors extracted from medical knowledge and the knowledge vectors extracted from neural networks, the features of the knowledge vectors we use are the intersection of features extracted from deep learning and medical knowledge. Therefore, we build the knowledge explainer to obtain heart disease features with and without ECG data.

With the knowledge explainer, we can get a knowledge vector for each class. Given an data set $\mathcal{D}_{ay} = \{(\mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$ with knowledge vector $\mathbf{a}^{(i)} \in \mathbb{R}^r$ and ECG label $y^{(i)} \in \{\mathbb{1}, \ldots, \mathbb{k}\}$. The output of the knowledge explainer, the knowledge vector, is mapped to medical knowledge, which enables the unseen categories to be given

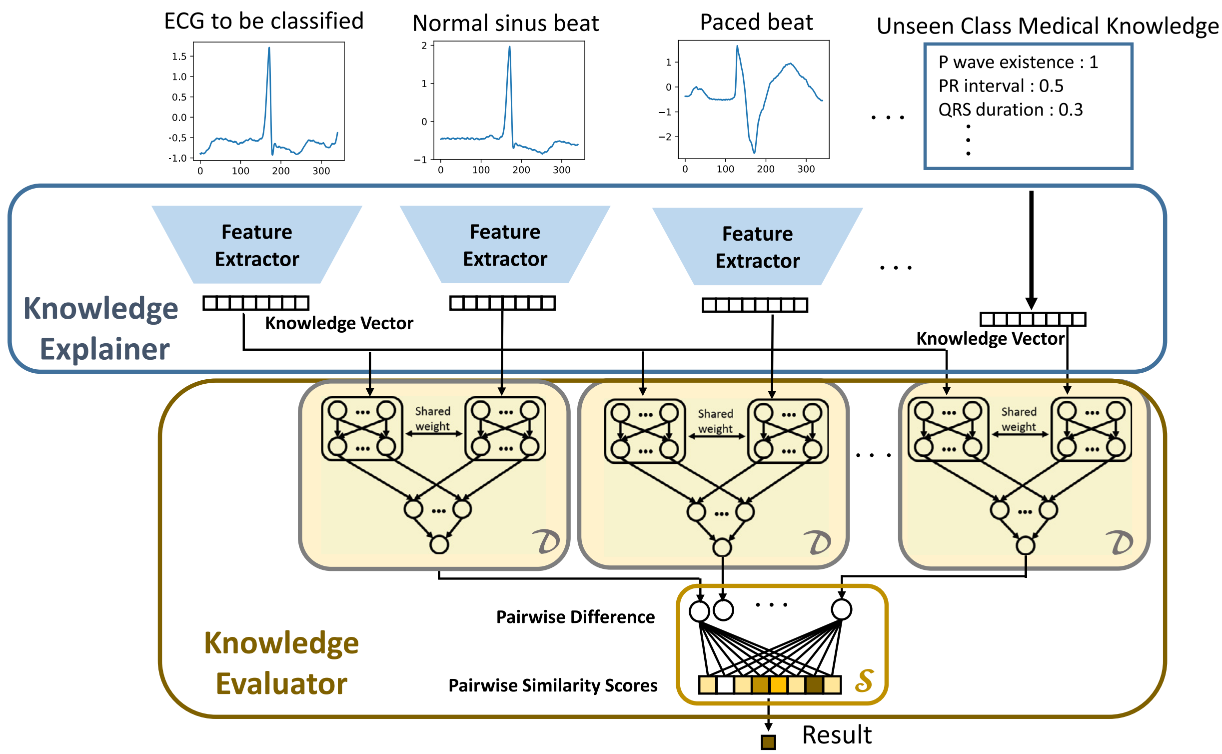


Fig. 1. The framework of KSL.

artificially in the middle layer of the model, enables the model to classify unseen categories, and makes the model interpretable.

3.3. Knowledge evaluator

Most of the existing neural network-based ECG classification methods treated each sample equally when training the model. In this case, the model is biased by rare classes and dominated by majority classes; then, rare classes would be more easily wrongly classified when the number of each category of the data set is imbalanced; the model would tend to weigh more on majority classes. For example, when there are 9997 samples of normal category, 2 samples of ventricular escape beat and 1 sample of atrial escape beat, most models learn to treat all samples as normal, which have a very high accuracy of 0.9997, but this model cannot detect ventricular escape beat and atrial escape beat.

In general, resampling strategies are often used to alleviate data imbalances. But, when the sample size of rare disease is too small, the model can not learn the characteristics of the disease. Besides, the knowledge vector corresponds to medical knowledge, which limits the model's ability to learn other features, which may lead to reduced performance, and the imbalance problem has not been solved.

To solve this problem, the classification method of our knowledge evaluator is to use the knowledge vector obtained by the knowledge explainer to calculate the similarity between the samples to be classified and the samples of each category and then select the most similar category instead of direct classification. In this way, our optimization goal is to distinguish whether two samples are similar. When encountering a sample of a rare category, the sample of a rare category can be classified correctly by judging the similarity, although it may not have learned the characteristics of the category. This prevents the model from being dominated by the majority class and allows classes with very few samples to be identified. It also enables the knowledge evaluator to obtain better performance in the case of using only knowledge vector classification that can correspond to medical knowledge.

We introduce the knowledge evaluator into two steps. First, we modify the target of $\mathcal{D}_\varphi$ to measure the difference of the paired ECG data, and then use $\mathcal{S}_\phi$ to calculate pairwise similarity to determine the ECG category instead of directly calculating the probability.

The first step $\mathcal{D}_\varphi$ of the knowledge evaluator is used to compute difference vector between knowledge vectors $\mathbf{a_i}$ and $\mathbf{a_j}$ using Eq. 4:

$$\hat{\mathbf{d}}^{(ij)} = \mathcal{D}_\varphi(\mathbf{a}^{(i)}, \mathbf{a}^{(j)}) \tag{4}$$

For $\mathcal{D}_\varphi$, we parameterized it as a neural network shared weights among paired ECG data. The neural network simultaneously receives 2 knowledge vector corresponds to pairwise ECG data. Then, it transforms the pair of input into a representation space of distance.

For the ECG classification problem with k diseases, we use $\mathbf{d}_c^{(i)}$ to represent the difference vector between the unknown knowledge vector $\mathbf{a}^{(i)}$ and the random sample of knowledge vector $\mathbf{a}_c$ for class c.

$$\hat{\mathbf{d}}_c^i = \mathcal{D}_\varphi(\mathbf{a}^{(i)}, \mathbf{a}_c) \tag{5}$$

The second step $\mathcal{S}_\phi$ of the knowledge interpreter is used to measure similarity between each class. We use $s_c^{(i)}$ to indicate whether the $\mathbf{a}^{(i)}$ and the random knowledge vector $\mathbf{a}_c$ in class c are the same class in Eq. 6. The pairwise similarity ground-truth $s_c^{(i)}$ would be 1 if $\mathbf{a}^{(i)}$ and $\mathbf{a}_c$ are in the same class, otherwise would be 0.

$$s_c^{(i)} = \begin{cases} 0 & , y^{(i)} \neq c \\ 1 & , y^{(i)} = c \end{cases} \tag{6}$$

Formally, the objective of the similarity scores with each class samples is computed using Eq. 7

$$\hat{s}_0^{(i)}, \ldots, \hat{s}_k^{(i)} = \mathcal{S}_\phi(\hat{\mathbf{d}}_0^{(i)}, \ldots, \hat{\mathbf{d}}_k^{(i)}) \tag{7}$$

We use $\mathcal{S}_\phi$ to calculate the similarity between the ECG to be classified and the ECG of each category. To compute ϕ and φ, we first randomly initialize the parameters of feature extractor ϕ and φ, and then optimize it by gradient descent as follows,

$$[\phi,\varphi] =: [\phi,\varphi] - \beta\nabla\mathcal{L}_{\mathcal{D}_{ay}}([\phi,\varphi]) \tag{8}$$

$$\mathcal{L}_{\mathcal{D}_{ay}}([\phi,\varphi]) = \frac{1}{|\mathcal{D}_{ay}|}\sum_{(\mathbf{a},y)\in\mathcal{D}_{ay}} l(\mathrm{argmax}(\mathcal{S}_\phi(\mathcal{D}_\varphi(\mathbf{a},\mathbf{a}_0),\ldots,\mathcal{D}_\varphi(\mathbf{a},\mathbf{a}_k))),y) \tag{9}$$

where l denotes the empirical loss which can be chosed as cross-entropy loss and β the learning rate.

The input of $\mathcal{S}_\phi$ is the difference vector of each class, and the output is the similarity of each class, instead of metric learning, just input the difference vector of one class and output the similarity of one class. This enables $\mathcal{S}_\phi$ to consider the difference between the input of unknown categories and the input of each category to give the overall similarity result, thereby improving overall performance. Since the inputs of $\mathcal{S}_\phi$ are diff vectors of all categories, the calculation of $\mathcal{S}_\phi$ is the calculation of the same deep learning architecture * # of class.

3.4. Training KSL

For training KSL, we propose to use a dual loop training process, as shown in Fig. 2. The detailed learning algorithm can be seen in Algorithm 1. We divide the KSL training algorithm into three loops. In the first loop, we use data set $\mathcal{D}_{xa}$, including knowledge vectors and labels, to optimize feature extractor $\mathcal{F}_\theta$ by Eqs. 3 and 2, so that ECG can be converted to knowledge vector well. In the second loop, we use data set $\mathcal{D}_{ay}$, including knowledge vectors data and labels, to optimize $\mathcal{S}_\phi$ and $\mathcal{D}_\varphi$ in knowledge evaluator by Eqs. 9 and 8, so that knowledge vector can be classified well. In the third loop, we use dataset $\mathcal{D}_{xy}$, including ECG data and labels, to optimize $\mathcal{F}_\theta, \mathcal{S}_\phi$ and $\mathcal{D}_\varphi$, so that KSL can optimize classification more directly. To compute θ, ϕ and φ, we optimize them by gradient descent as follows,

$$[\theta,\phi,\varphi] =: [\theta,\phi,\varphi] - \gamma\nabla\mathcal{L}_{\mathcal{D}_{xy}}([\theta,\phi,\varphi]) \tag{10}$$

$$\mathcal{L}_{\mathcal{D}_{xy}}([\theta,\phi,\varphi]) = \frac{1}{|\mathcal{D}_{xy}|}\sum_{(x,y)\in\mathcal{D}_{xy}} l(\mathrm{argmax}(\mathcal{S}_\phi(\mathcal{D}_\varphi(\mathcal{F}_\theta(x),\mathcal{F}_\theta(x^0)),\ldots,\mathcal{D}_\varphi(\mathcal{F}_\theta(x),\mathcal{F}_\theta(x^k)))),y) \tag{11}$$

where l denotes the following empirical loss, e.g. cross-entropy loss and γ denotes the learning rate.

Dividing the network into two parts (knowledge explainer and knowledge evaluator) for training can add some categories that are not in the training data through the knowledge vector in the middle of the medical knowledge, so that the model can separate the categories that do not appear in the training data set. In addition, the training goal of the knowledge explainer is to obtain a knowledge vector that can correspond to medical knowledge, so dividing the network into two parts can increase interpretability. However, the method of dividing the network into two parts and training separately has no way to optimize the entire model with the final classification result, which may cause poor classification results. Therefore, during training, an outer loop is added, which can optimize the final classification result of the entire network and improve the overall performance, but it cause the calculation amount of the model to be doubled during the training.

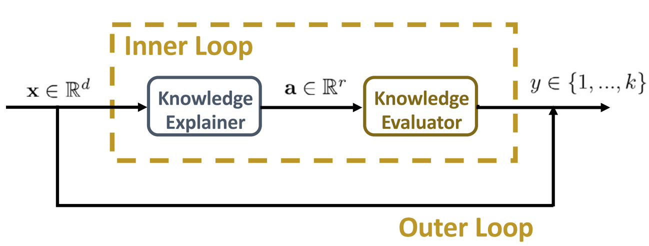


Fig. 2. Dual loop training strategy in KSL.

Algorithm 1: Training KSL

Input
$\mathcal{D}_{xy} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$
$\mathcal{D}_{xa} = \{(\mathbf{x}^{(i)}, a^{(i)})\}_{i=1}^n$
$\mathcal{D}_{ay} = \{(\mathbf{a}^{(i)}, y^{(i)})\}_{i=1}^n$
Output
Feature extractor $\mathcal{F}_\theta$ with parameter θ
Pairwise similarity calculator $\mathcal{S}_\phi$ with parameterϕ
Pairwise difference calculator $\mathcal{D}_\varphi$ with parameter φ

```
Randomly initialize θ, ϕ, and φ
for each epoch
  # Inner Loop
  for samples in D_xa
    Evaluate L_{D_xa}(θ) by Eq. 3
    Optimize θ by Eq. 2
  end for
  for samples in D_ay
    Evaluate L_{D_ay}([ϕ, φ]) by Eq. 9
    Optimize [ϕ, φ] by Eq. 8
  end for
  # Outer Loop
  for samples in D_xy
    Evaluate L_{D_xy}([θ, ϕ, φ]) by Eq. 11
    Optimize [θ, ϕ, φ] by Eq. 10
  end for
end for
```

4. Interpretable results from KSL

Interpretable results aim to provide necessary explanations about given results that can help a human understand "why is it" beyond "what is it". In the medical field, the method of computer-aided diagnosis requires a high degree of interpretability so that doctors and patients have a high degree of trust in the method.

The main purpose of interpretability techniques is to reveal the prediction process of the machine learning model and to explain the results of the model predictions. We divide the current interpretability research into two categories: (1) directly building an interpretable model, e.g. [22,17] used the attention mechanism to explain the relationship between outcome and input ECG. (2) interpreting black-box model with a simpler proxy model, e.g. SHAP [33] uses an idea of game theory to measure the role of each feature in the prediction process, and LIME [34] gives interpretability by adding perturbations to input instances and changes in results. In this paper, we integrate such human understandable part (knowledge vector) into deep models. Below we discuss the interpretability of the knowledge interpreter and the knowledge evaluator separately.

In the knowledge explainer, we can get the knowledge vector in two ways. The first way is using a feature extractor to convert the ECG into the knowledge vector, which corresponds to medical knowledge that can be understood by a human. The second way is to artificially give knowledge vectors of unseen classes using the golden role of medicine. The output of the knowledge explainer, the knowledge vector, has the same meaning as what human's knowledge. As the middle layer of the whole model, knowledge vector can correspond to medical knowledge that humans can understand. This shows that the classification method of our model is based on the knowledge vector that people can understand to give the classification results. Therefore, KSL is interpretable.

In the knowledge evaluator, we compare the knowledge vectors to be classified with the knowledge vectors of each category, and select the closest category instead of using the knowledge vectors to classify directly. This makes it easier for people to understand the process of classification decision-making, so knowledge evaluator is more explanatory than using knowledge vector to classify directly.

To sum up, we use the knowledge vector corresponding to human knowledge in the middle of the whole model and use pairwise similarity between categories to make the final classification, so that KSL is interpretable.

5. Experiment

5.1. Experimental settings

Dataset We use the MIT-BIH Arrhythmia Database [35] which is a widely used real-world ECG dataset. The dataset consists of 48 records, each containing a 30 min ECG segment selected from a 24-h record of 48 different patients. Each ECG signal is an 11-bit resolution over a 10 mV range with a sampling frequency of 360 Hz. We extract ECG beats by cutting 1 s around the position of the QRS complex, which has been marked in the dataset. The statistics of the dataset is shown in Table 2. Our task is to classify every ECG beat into one of the 12 classes.

Implementation Details For KSL, we extract following expert features and compose them to train the knowledge explainer: P wave existence (1 for existence, 0 for not), P wave amplitude (unit is mV), QRS complex amplitude (unit is mV), T wave amplitude (unit is mV), PR interval (unit is second), QRS duration (unit is second), QT interval (unit is second), and ST slope (unit is mV/s). We show the architecture of feature extractor $\mathcal{F}$ in Table 3, and the architecture of the knowledge evaluator shown in Table 4. KSL was optimized using Stochastic gradient descent (SGD), with the learning rate set to 0.001. For preprocessing, we use MinmaxScaler to translate ECG data between 0 and 1, and use StandardScaler to standardize attributes by removing the mean and scaling to unit variance. All methods were implemented in pytorch version 1.1.0 and scikit-learn [36], trained on a system equipped with 64 GB RAM, 12 Intel Core i7-6850 K 3.60 GHz CPUs and Nvidia GeForce GTX 1060.

Measurements We use macro-F_1 score to measure classification performance. F_1-score is the harmonic average of the precision and recall $F_1 = (2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall})$, where Precision is the ratio of predicted positives that are actually positive samples in the ground truth, and Recall is the ratio of positives that are correctly classified to actually positive samples. The macro-F_1 score is the average F_1 score of each class. We use macro-F_1 score because it is a more comprehensive measurement.

5.2. Comparing with baselines

The compared methods include:

- **LR**: We use the same knowledge explainer as KSL, but replace the knowledge evaluator to be a Logistic Regression (LR).
- **RF**: We use the same knowledge explainer as KSL, but replace the knowledge evaluator to be a Random Forest (RF) [37]. The number of trees in RF is set to 20.
- **Nearest**: We use the same knowledge explainer as KSL, but replace the knowledge evaluator to be the nearest neighbor classifier (Nearest), the number of neighbors is set to 8.
- **NN**: We use the same knowledge explainer as KSL, but replace the knowledge evaluator to a neural network. We weight more on rare classes by oversampling them to get more samples.
- **CNN**: We build the CNN model with three 1-d convolutional layers based on [38]. Each layer has 64 filters with kernel size set to 3 and stride set to 2, using ReLU as activation function and SGD for optimization with oversampling [39].

As shown in Table 2, we take 50 test sets from each category (less than 50 categories of "e", we take 8 as test sets) and the rest as training sets. For each compared method, we run 10 times and report the average number (blue bar) as well as the standard deviation (black error bar) as the confidence interval of each method. The macro-F_1 score of all categories is shown in Fig. 3. We can see that the performance of using neural networks is better than that of using traditional machine learning methods, and KSL performed better than all baseline in ECG data with imbalanced categories.

In order to show the effect of extreme imbalance on performance, the training samples are sampled from 2, 4, 8, 16 of the least five categories, and the training samples from other categories remain unchanged. We only test the macro-F_1 scores of the least five categories in our test set, and the result shows in Fig. 4. It shows that KSL's performance in a few sample categories exceeds all comparison methods. As the amount of data decreases, the performance difference between KSL and other deep learning methods becomes larger. This can indicate that KSL is more able to distinguish a few categories than other methods in the case of

Table 2
Data profiles of MIT-BIH dataset.

Label	Description	# of Samples	Proportion	# of Test Samples
N	Normal sinus beat	74749	71.23%	50
L	Left bundle branch block beat	8071	7.69%	50
R	Right bundle branch block beat	7255	6.92%	50
V	Premature ventricular contraction	7123	6.79%	50
/	Paced beat	3619	3.45%	50
A	Atrial premature beat	2546	2.43%	50
F	Fusion of ventricular and normal beat	802	0.76%	50
f	Fusion of paced and normal beat	260	0.25%	50
j	Junctional escape beat	229	0.22%	50
a	Aberrated atrial premature beat	150	0.14%	50
E	Ventricular escape beat	106	0.10%	50
e	Atrial escape beat	16	0.02%	8
Total		104926	100%	558

Table 3
Detailed architecture of the feature extractor $\mathcal{F}$.

Layer	Type	Kernel size	Activation	Nonlinearity
0	Input	(1,360)		
1	Convolution	(1,3)		
2	Max Pooling	(1,2)	ReLU	Batch Normalization
3	Convolution	(1,3)		
4	Max Pooling	(1,2)	ReLU	Batch Normalization
5	Convolution	(1,3)		
6	Max Pooling	(1,2)	ReLU	Batch Normalization
7	Fully Connected	11008	ReLU	
8	Fully Connected	500	Sigmoid	
9	Fully Connected	Number of Attribute		

Table 4
Detailed architecture of the knowledge evaluator $\mathcal{G}$.

Layer	Type	Kernel size	Activation
0	Input	8	
1	Fully Connected	64	ReLU
2	Fully Connected	128	ReLU
3	Fully Connected	256	ReLU
4	Concatenate	256 * k	ReLU
5	Fully Connected	256 * k	ReLU
6	Fully Connected	64 * k	ReLU
7	Fully Connected	k	

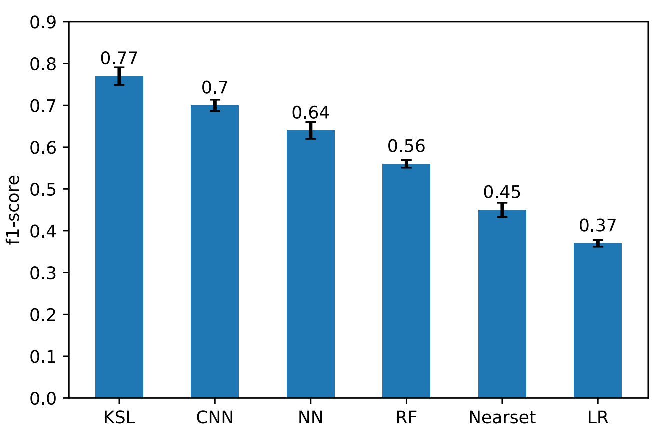


Fig. 3. Comparison of model performance averaging over all labels. The blue bars show the average numbers and the black error bars show the standard deviations.

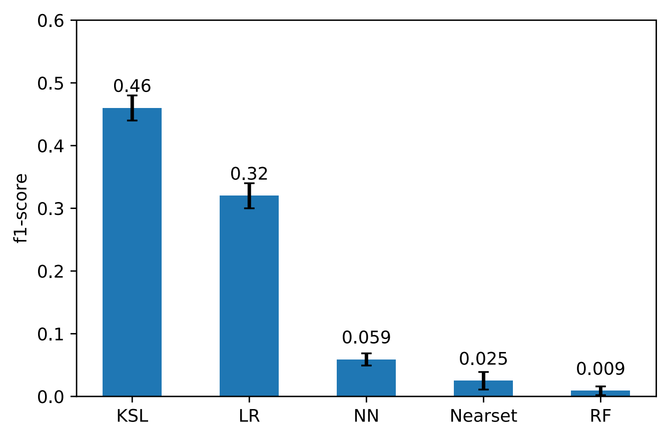


Fig. 5. Comparison of model performance for classifying unseen ECG data. The blue bars show the average numbers averaging over f, j, a, E, and e. The black error bars show the standard deviations.

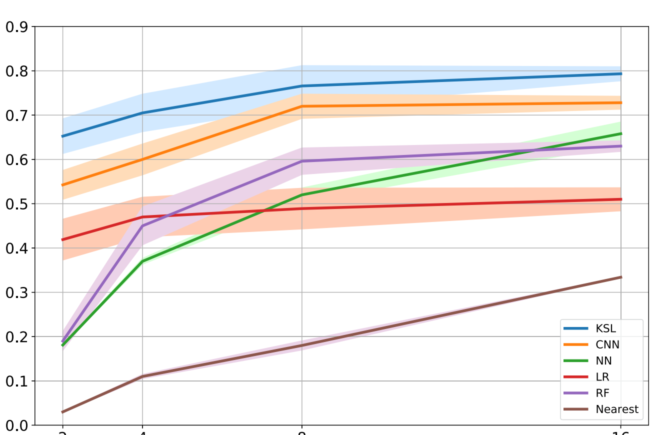
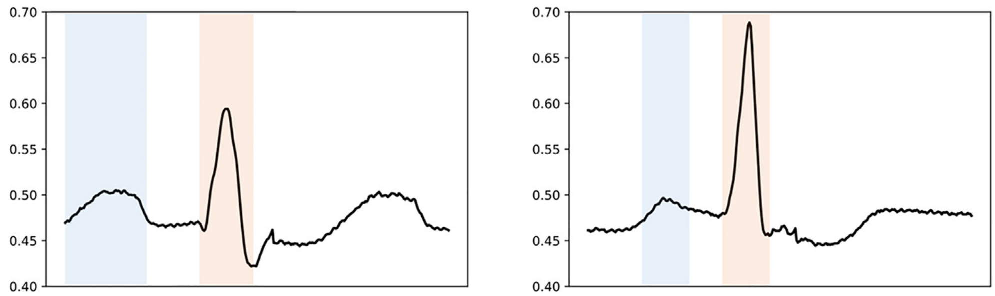


Fig. 4. Comparison of model performance averaging over f, j, a and E. The x-axis shows different number of training samples used in each method. The solid lines show the average numbers and the shades show the standard deviations.

Table 5
Error rate of knowledge vector.

Attribute	Error rate
P wave existence	0.366 ± 0.002
P wave amplitude	0.020 ± 0.001
QRS complex amplitude	0.007 ± 0.000
T wave amplitude	0.013 ± 0.000
PR interval	0.130 ± 0.001
QRS duration	0.019 ± 0.000
QT interval	0.006 ± 0.000
ST slope	0.009 ± 0.000

imbalanced data. LR is not affected by the sample size, but the overall accuracy is poor. When the sample size of RF, NN and Nearest increased from 2 to 16, the macro-F_1 scores increased significantly. Nearest is the worst performer in extremely unbalanced data. We believe that when looking for the nearest neighbor, the way to calculate distance is to directly calculate the L2 distance



Expert Knowledge	Left ECG	Right ECG
P wave existence	**0.8825**	0.1068
P wave amplitude	**0.2626**	0.0039
QRS complex amplitude	0.8837	**1.8464**

Fig. 6. The interpretation results of the knowledge explainer in KSL. We can see that the knowledge explainer gives important expert knowledge without the help of human experts. Compared with the right ECG, the left ECG has more clear and wider P wave but has less significant QRS complex. This evidence is reflected by higher P wave existence probability, higher P wave amplitude, and lower QRS complex amplitude, as shown in the table.

by treating each feature as equally important. Therefore, when the amount of data in this category is reduced, it is not easy to have samples that are very similar to the samples to be classified in the training data set, which lead to poor classification results. In this experiment, we set number of neighbors in Nearest to 1, 2, 4, 8 separately.

5.3. Classifying unseen ECG data

In this experiment, we show the ability of KSL on classifying unseen ECG data when providing necessary medical domain knowledge. Specifically, we hide the ECG data with classes f, j, a, E, and e when training KSL, provided values of knowledge vector from golden rules, and then classify these unseen ECG data based on the approach. The comparison method is LR, NN, RF, Nearest, which we provide the same necessary medical domain knowledge, but compare it with test ECG directly without knowledge evaluator. Results are shown in Fig. 5. Although the macro-F_1 score of KSL = 0.428 is not good enough, it is still better than other methods. LR is not outstanding in overall performance(in Fig. 3), but its performance is relatively unaffected in the case of extreme imbalance and unseen classes. NN is affected by data imbalance (in Table 4), but it still has a certain degree of discrimination in invisible categories.

5.4. Interpretable results

To show the interpretation results of the knowledge explainer, we list the error rates of all the attributes used in the experiments in Table 5. From the table we can see that the error rate of all features is very small, and the knowledge vector, as the middle layer of KSL, can well correspond to medical knowledge.

We randomly select some of ECG data along with their learned knowledge vector, to visually check whether they are consistent or not. Results are shown in Fig. 6. We can see that the knowledge explainer gives important expert knowledge without the help of human experts. Compared with the right ECG, the left ECG has more clear and wider P wave but has less significant QRS complex. This evidence is reflected by higher P wave existence probability, higher P wave amplitude, and lower QRS complex amplitude, as shown in the table below.

6. Limitations

KSL has a higher time complexity than traditional neural network. The reason is that KSL needs to calculate the difference between the sample and each type of electrocardiogram through the knowledge explainer, and then concatenate all the differences to become the input of the knowledge evaluator. In detail, we use $c(\mathcal{H}\prime)$ and $c(\mathcal{G}\prime)$ represent the same deep learning architecture computation of knowledge explainer and knowledge evaluator, respectively. In the case of k classification, the computation of knowledge explainer $c(\mathcal{H}) = c(\mathcal{H}\prime) \times k$ and the computation knowledge evaluator $c(\mathcal{G}) \geqslant c(\mathcal{G}\prime) \times k^2$. Besides, KSL can classify unseen classes only if the knowledge vector of these class is artificially given.

7. Conclusion and future work

In this paper, we propose an interpretable model named KSL to classify imbalanced ECG data. KSL can be effectively trained on imbalanced ECG data; it can even classify unseen diseases if provided necessary medical domain knowledge. Besides, KSL also provides simple and clear interpretable explanations along with classification results. These interpretable explanations are easily understandable for cardiologists, thus can be directly used in the medical area. Experiments on real-world ECG data show that KSL improves 10.00% of F_1-score on imbalanced classes, and 43.75% of F_1-score on unseen classes, compared with the second-best baseline.

In the future, we plan to investigate advanced techniques to reduce the time complexity of KSL. Besides, we will also study on deep learning methods on ECG for solving other new emerging interdisciplinary topics such as biometric identification, cardiac reaction time in safe driving and so on.

CRediT authorship contribution statement

Yen-hsiu Chou: Conceptualization, Methodology, Validation, Investigation, Writing - original draft. **Shenda Hong:** Conceptualization, Data curation, Writing - review & editing. **Yuxi Zhou:** Visualization, Writing - review & editing. **Junyuan Shang:** Software, Writing - review & editing. **Moxian Song:** Formal analysis, Validation. **Hongyan Li:** Supervision.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] E.J. Benjamin, S.S. Virani, C.W. Callaway, A.M. Chamberlain, A.R. Chang, S. Cheng, S.E. Chiuve, M. Cushman, F.N. Delling, R. Deo, et al., Heart disease and stroke statistics-2018 update: a report from the american heart association, Circulation 137 (12) (2018) e67–e492.

[2] S.P. Keller, H.R. Halperin, Cardiac arrest: the changing incidence of ventricular fibrillation, Curr. Treat. Options Cardiovasc. Med. 17 (7) (2015) 29.

[3] H. Pürerfellner, E. Pokushalov, S. Sarkar, J. Koehler, R. Zhou, L. Urban, G. Hindricks, P-wave evidence as a method for improving algorithm to detect atrial fibrillation in insertable cardiac monitors, Heart Rhythm 11 (9) (2014) 1575–1583.

[4] M. Carrara, L. Carozzi, T.J. Moss, M. De Pasquale, S. Cerutti, M. Ferrario, D.E. Lake, J.R. Moorman, Heart rate dynamics distinguish among atrial fibrillation, normal sinus rhythm and sinus rhythm with frequent ectopy, Physiol. Measure. 36 (9) (2015) 1873.

[5] A. Petrenas, V. Marozas, L. Sörnmo, A. Lukosevicius, An echo state neural network for qrst cancellation during atrial fibrillation, IEEE Trans. Bio-Med. Eng. 59 (10) (2012) 2950.

[6] S. Ladavich, B. Ghoraani, Rate-independent detection of atrial fibrillation by statistical modeling of atrial activity, Biomed. Signal Process. Control.

[7] J.-S. Wang, W.-C. Chiang, Y.-L. Hsu, Y.-T.C. Yang, Ecg arrhythmia classification using a probabilistic neural network with a feature reduction method, Neurocomputing 116 (2013) 38–45.

[8] S. Hong, Y. Zhou, J. Shang, C. Xiao, J. Sun, Opportunities and challenges in deep learning methods on electrocardiogram data: a systematic review, arXiv preprint arXiv:2001.01550.

[9] G.D. Clifford, C. Liu, B. Moody, I. Silva, Q. Li, A. Johnson, R. Mark, AF Classification from a Short Single Lead ECG Recording: The PhysioNet Computing in Cardiology Challenge 2017, CinC 44.

[10] Z. Xiong, M.K. Stiles, J. Zhao, Robust ecg signal classification for detection of atrial fibrillation using a novel neural network, CinC 2017 (44) (2017) 1.

[11] A.Y. Hannun, P. Rajpurkar, M. Haghpanahi, G.H. Tison, C. Bourn, M.P. Turakhia, A.Y. Ng, Cardiologist-level Arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network, Nat. Med. 25 (1) (2019) 65.

[12] T. Teijeiro, C.A. Garcia, P. Felix, D. Castro, Abductive reasoning as a basis to reproduce expert criteria in ECG atrial fibrillation identification, Physiol. Measure. 39.

[13] P. Schwab, G.C. Scebba, J. Zhang, M. Delai, W. Karlen, Beat by beat: classifying cardiac Arrhythmias with recurrent neural networks, CinC 2017 (44) (2017) 1.

[14] R. Salloum, C.-C.J. Kuo, ECG-based biometrics using recurrent neural networks, in: ICASSP, 2017 IEEE International Conference on, IEEE, 2017, pp. 2062–2066.

[15] C. Zhang, G. Wang, J. Zhao, P. Gao, J. Lin, H. Yang, Patient-specific ECG classification based on recurrent neural networks and clustering technique, in: Biomedical Engineering (BioMed), 2017 13th IASTED International Conference on, IEEE, 2017, pp. 63–67.

[16] Y. Zhou, S. Hong, J. Shang, M. Wu, Q. Wang, H. Li, J. Xie, K-margin-based residual-convolution-recurrent neural network for atrial fibrillation detection, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 6057–6063.

[17] S. Hong, C. Xiao, T. Ma, H. Li, J. Sun, Mina: multilevel knowledge-guided attention for modeling electrocardiography signals, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, AAAI Press, 2019, pp. 5888–5894.

[18] Y. Li, Y. Pang, J. Wang, X. Li, Patient-specific ecg classification by deeper cnn from generic to dedicated, Neurocomputing 314 (2018) 336–346.

[19] X. Dong, C. Wang, W. Si, Ecg beat classification via deterministic learning, Neurocomputing 240 (2017) 1–12.

[20] S. Poungponsri, X.-H. Yu, An adaptive filtering approach for electrocardiogram (ecg) signal noise reduction using neural networks, Neurocomputing 117 (2013) 206–213.

[21] H.B. Bae, M.S. Park, R.M. Kil, H.Y. Youn, Classifying heart conditions based on class probability output networks, Neurocomputing 360 (2019) 198–208.

[22] Y. Xu, S. Biswal, S.R. Deshpande, K.O. Maher, J. Sun, Raim: recurrent attentive and intensive model of multimodal patient monitoring data, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 2565–2573.

[23] S. Hong, M. Wu, Y. Zhou, Q. Wang, J. Shang, H. Li, J. Xie, Encase: An ensemble classifier for ecg classification using expert features and deep neural networks, in: Computing in Cardiology, 2017, IEEE, 2017, pp. 1–4.

[24] S. Hong, Y. Zhou, M. Wu, J. Shang, Q. Wang, H. Li, J. Xie, Combining deep neural networks and engineered features for cardiac arrhythmia detection from ecg recordings, Physiol. Measure. 40 (5) (2019) 054009.

[25] S.P. Shashikumar, A.J. Shah, Q. Li, G.D. Clifford, S. Nemati, A deep learning approach to monitoring and detecting atrial fibrillation using wearable technology, in: BHI, 2017 IEEE EMBS International Conference on, IEEE, 2017, pp. 141–144.

[26] Z. Golrizkhatami, A. Acan, Ecg classification using three-level fusion of different feature descriptors, Expert Syst. Appl. 114 (2018) 54–64.

[27] S.M. Al-Khatib, W.G. Stevenson, M.J. Ackerman, W.J. Bryant, D.J. Callans, A.B. Curtis, B.J. Deal, T. Dickfeld, M.E. Field, G.C. Fonarow, et al., 2017 aha/acc/hrs guideline for management of patients with ventricular arrhythmias and the prevention of sudden cardiac death: a report of the american college of cardiology/american heart association task force on clinical practice guidelines and the heart rhythm society, J. Am. College Cardiol. 72 (14) (2018) e91–e220.

[28] Standard, ANSI/AAMI EC57/Ed.3, Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms.

[29] Z.C. Lipton, The mythos of model interpretability, Queue 16 (3) (2018) 31–57.

[30] E. Ahmed, M. Jones, T.K. Marks, An improved deep learning architecture for person re-identification, in, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3908–3916.

[31] A. Hermans, L. Beyer, B. Leibe, In defense of the triplet loss for person re-identification, CoRR. URL:http://arxiv.org/abs/1703.07737.

[32] S. Bak, P. Carr, One-shot metric learning for person re-identification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2990–2999.

[33] S.M. Lundberg, S.-I. Lee, A unified approach to interpreting model predictions, in: Advances in Neural Information Processing Systems, 2017, pp. 4765–4774.

[34] M.T. Ribeiro, S. Singh, C. Guestrin, why should i trust you? Explaining the predictions of any classifier, in, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144.

[35] A.L. Goldberger, L.A. Amaral, L. Glass, J.M. Hausdorff, P.C. Ivanov, R.G. Mark, J.E. Mietus, G.B. Moody, C.-K. Peng, H.E. Stanley, Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals, Circulation 101 (23) (2000) e215–e220.

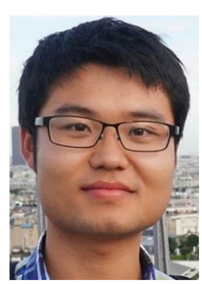
[36] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, J. Mach. Learn. Res. 12 (Oct) (2011) 2825–2830.

[37] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.

[38] S. Kiranyaz, T. Ince, M. Gabbouj, Real-time patient-specific ecg classification by 1-d convolutional neural networks, IEEE Trans. Biomed. Eng. 63 (3) (2015) 664–675.

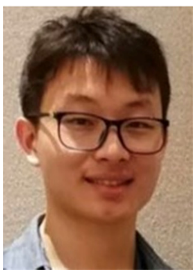
[39] G. Lemaıtre, F. Nogueira, C.K. Aridas, Imbalanced-learn: a python toolbox to tackle the curse of imbalanced datasets in machine learning, J. Mach. Learn. Res. 18 (1) (2017) 559–563.

Yen-hsiu Chou received the B.S. Degree in Computer Science from National Tsing Hua University, in 2018. She is currently pursuing the M.S. degree in School of Electronics Engineering and Computer Sciences at Peking University. Her current research interests are machine learning and data mining for healthcare.



Shenda Hong is a PostDoc of College of Computing at Georgia Institute of Technology. Prior to that, he obtained his Ph.D. degree in School of Electronics Engineering and Computer Sciences at Peking University in 2019, and B.S. degree in School of Science at Beijing University of Posts and Telecommunications in 2014. His research interests are machine learning and data mining for healthcare, especially deep learning methods on Electronic Health Records and biomedical signals like Electrocardiogram and Electroencephalogram.

Yuxi Zhou obtained the M.E. degrees in School of Peking University, China, in 2013. She is currently pursuing the Ph.D. degree with the Key Laboratory of Machine Perception, Peking University, China. Her main research interests include machine learning, data mining, artificial intelligence for healthcare, especially intelligent diagnosis of Electrocardiogram (ECG) and Electronic Health Records (EHR).



Junyuan Shang is currently pursuing the M.S. degree in School of Electronics Engineering and Computer Sciences at Peking University. Prior to that, he obtained his B.S. Degree in Computer Science from Huazhong University of Science and Technology in 2017. His research interests are machine learning and deep learning models with medical and healthcare applications.

Moxian Song is a Ph.D candidate of School of Electronics Engineering and Computer Science at Peking University. Prior to that, he obtained his B.S. degree in School of Electronics and Information at Northwestern Polytechnical University in 2018. His research interests are machine learning and data mining, especially partial label learning and information fusion.

Hongyan Li is a professor in the Department of Intelligence Science, School of EECS. She obtained her B.Eng. and Ph.D. from Northwestern Polytechnical University in 1993 and 1999 respectively. Her research interests include data management, business process control, data mining and knowledge discovery, with a focus on big data management, large-scale data analysis, and business process anomalies detecting. She has served in Database Specialized Committee of China Computer Federation and Cloud Computing & SaaS Specialized Committee of China Institute of Communications.