Neural Computing and Applications
https://doi.org/10.1007/s00521-021-06219-9

SPECIAL ISSUE ON IOT-BASED HEALTH MONITORING SYSTEM

Toward real-time and efficient cardiovascular monitoring for COVID-19 patients by 5G-enabled wearable medical devices: a deep learning approach

Liang Tan[1,2] · Keping Yu[3] · Ali Kashif Bashir[4,5] · Xiaofan Cheng[1] · Fangpeng Ming[1] · Liang Zhao[6] · Xiaokang Zhou[7]

Received: 12 October 2020 / Accepted: 8 June 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract
Patients with deaths from COVID-19 often have co-morbid cardiovascular disease. Real-time cardiovascular disease monitoring based on wearable medical devices may effectively reduce COVID-19 mortality rates. However, due to technical limitations, there are three main issues. First, the traditional wireless communication technology for wearable medical devices is difficult to satisfy the real-time requirements fully. Second, current monitoring platforms lack efficient streaming data processing mechanisms to cope with the large amount of cardiovascular data generated in real time. Third, the diagnosis of the monitoring platform is usually manual, which is challenging to ensure that enough doctors online to provide a timely, efficient, and accurate diagnosis. To address these issues, this paper proposes a 5G-enabled real-time cardiovascular monitoring system for COVID-19 patients using deep learning. Firstly, we employ 5G to send and receive data from wearable medical devices. Secondly, Flink streaming data processing framework is applied to access electrocardiogram data. Finally, we use convolutional neural networks and long short-term memory networks model to obtain automatically predict the COVID-19 patient's cardiovascular health. Theoretical analysis and experimental results show that our proposal can well solve the above issues and improve the prediction accuracy of cardiovascular disease to 99.29%.

Keywords 5G · Cardiovascular monitoring · Deep learning · Flink · CNN · LSTM

1 Introduction

COVID-19 is the first global coronavirus pandemic that humanity has ever faced, and awareness of COVID-19 is still growing. In addition to advanced medical technology, big data plays a crucial role in the prevention and control of COVID-19 [1, 2]. In the real-time environment constituted by the Internet of Things (IoT), not only sensors and mobile devices have generated a large amount of data, but also software applications, web, and other resources have generated massive data [3, 4]. Especially in medical and healthcare, due to the emergence of a large number of wearable medical devices, the data generated by these devices need to be collected remotely in real time. However, it brought many new challenges. One of the most critical challenges is to extract streaming data in real time and process and analyze different types of data [5].

The death of COVID-19 patients is often accompanied by underlying cardiovascular and other diseases. Cardiovascular diseases characterized by suddenness have become one of the main diseases threatening human health [6]. A large number of medical practices show that when the heart suddenly stops, the best rescue time is within 4 minutes. At the same time, we found that if we can detect subtle signs in advance and take effective measures, 70% of cardiovascular disease patients, including heart attacks, can avoid death. Currently, electrocardiogram (ECG) is the most straightforward and efficient clinical examination method for all kinds of cardiovascular and cerebrovascular diseases. Therefore, real-time and effective monitoring, analysis, and diagnosis of the COVID-19 patient's ECG signal are particularly important [7]. Moreover, the communication between doctors and COVID-19 patients is not accurate, convenient, and timely, such as telephone and text communication. Although doctors and hospitals can observe the health of COVID-19 patients through

Extended author information available on the last page of the article

Published online: 04 July 2021

inspections, this traditional hospital-based diagnosis and treatment lack immediacy and continuity, and it is difficult to capture the COVID-19 patients' signs. Therefore, it is of practical significance to provide real-time ECG diagnosis and predict services for those in need [8].

With wearable medical devices (such as heart rate monitoring cuffs, blood pressure, blood glucose meters, etc.), it can continuously track COVID-19 patients' health and provide personalized healthcare solutions [9]. However, the data continuously generated by wearable medical devices need not only real-time processing but also diagnosis and prediction. Existing wireless communication technologies have problems such as high latency and low speed. Moreover, relying on manual recognition of electrocardiogram diagnosis methods can no longer meet the current medical needs. Simultaneously, traditional machine learning-based methods need to manually extract electrocardiogram features, which cannot realize the automation of the diagnosis process and the accuracy of the results. Therefore, it is a significant challenge to process a large amount of data generated by sensors and diagnose them in real time in a critical situation [10].

Given the aforementioned difficulties with traditional cardiovascular diseases diagnoses and wearable device data processing and analysis, the automatic classification and diagnosis of assisted ECG signals for COVID-19 patients based on deep learning is an effective solution to the above problems. In addition, with the emergence of 5G [11, 12], a solution with high throughput and low latency has been provided for processing a large amount of monitoring data [13]. Moreover, the development of big data technology also provides many open source platforms for real-time processing of streaming data, such as Spark, Druid, and Flink. With the continuous advancement of artificial intelligence technology, the application of artificial intelligence technology to medical diagnosis is a general trend [14]. In recent years, in order to meet the needs of high-speed and high-precision ECG analysis, deep neural networks have been widely used in automatic ECG diagnosis [15].

This article proposes a real-time cardiovascular monitoring system for COVID-19 patients based on 5G and deep learning with the assist of wearable medical devices that can transmit human ECG signal data. The Second 2 introduces related work. The overall architecture of the real-time monitoring system is proposed in Sect. 3. Section 4 is the ECG signal classification algorithm based on deep learning. In Sect. 5, we put forward the experimental method and evaluation. Finally, Sect. 6 summary the conclusion.

2 Related works

In recent years, big data analysis related to healthcare has become an important issue in many research fields, such as machine learning, deep learning, and data mining using medical and health data and information available in hospitals. The progress of the data collection process comes from the tremendous development of technology in the medical and health field, in which data records are collected through three main stages of digital data flow generated from patient clinical records, health research records and organization operations [16]. Analyzing these data for computer-aided diagnosis and then developing real-time systems has become the development trend of today's smart medical care.

Sun and Reddy et al. [17] discussed an overview of health care data sources. This research analyzes that health care data plays a very important role in many systems such as disease prediction, prevention methods, medical guidance, and emergency medical decision-making to improve health, reduce costs and increase efficiency. And in the existing research, a variety of Spark machine learning models have been used in medical databases. For example, in [18], a real-time health prediction system using spark machine learning streaming big data is introduced. The system is tested on tweets of users with health attributes. The system receives the tweets, extracts features, and uses decision tree algorithms to predict The health of the user and finally, sends the information directly to the user to take appropriate action. In addition, Alottaibi et al. [19] proposed a Sehaa-Kingdom of Saudi Arabia (KSA) Central Arab Healthcare Twitter data big data analysis tool. The system uses two different machine learning algorithms, including naive Bayes and logistic regression algorithms, and applies multiple feature extraction methods to detect various diseases in KSA. In [20], a system based on Apache Spark that can predict heart disease in real time uses memory computing to apply machine learning to streaming data. The system is divided into two stages. The first stage is to apply classification algorithms to data for heart disease prediction through Spark MLlib and Spark streaming, and the second stage is to use Apache Cassandra to store massively generated data and visualizations.

In addition, many systems that use wearable devices to collect data to predict cardiovascular disease have been proposed in recent years [21, 22]. Al-Makhadmeh and Tolba et al. [23] proposed a heart disease detection system based on wearable medical equipment. The system transmits the collected patient heart data to the medical system and then uses feature extraction technology and deep learning model Value feature extraction and correct classification. Lin et al. [24] proposed a system based on

Support Vector Machine (SVM) classifier to predict patients suffering from left ventricular hypertrophy. The system collects young people's age, height, weight and electrocardiogram data for rapid diagnosis. Khade et al. [25] proposed a cardiovascular disease prediction system based on SVM and Convolutional Neural Networks (CNN). The system sends ECG signals to the SVM classifier and Boosted Decision Tree to classify cardiovascular diseases and then uses CNN to predict severe Degree, and the accuracy rate is 88.3%. Zhao et al. [26] proposed a system that uses CNN to detect the original ECG signal in a wearable device. This system avoids the traditional manual feature extraction process and implements the system based on the cloud. In addition, Kumar et al. [27] proposed a three-layer framework system with an ML model to receive data from wearable devices and perform analysis and processing. The first layer collects ECG data from wearable devices. The second layer stores healthcare data in the cloud, and the third layer uses logistic regression algorithms to predict cardiovascular diseases.

Moreover, many researchers have used deep learning in the field of myocardial infarction detection and coronavirus detection. M Hammad et al. [28] proposed an automatic detection method for myocardial infarction based on the end-to-end model of a deep convolutional neural network and used the focal loss function optimization model for data imbalance, and finally reached 98.84% on the PTB data set. [29] provides a promising solution by proposing a COVID-19 detection system based on deep learning. The simulation results reveal that the proposed deep learning modalities can be considered and adopted for quick COVID-19 screening; [30] propose a modified version of Fuzzy C-Means for segmenting 3D medical volumes, which has been rarely implemented for 3D medical image segmentation; in [31], two new quantum information hiding methods are proposed for telemedicine image sharing. The results show that the proposed methods have excellent visual quality, high embedding capability and security.

Most researches rely on specific medical care data sources and apply them on offline systems. However, medical data sources are diverse and new data are constantly being produced. Real-time healthcare analysis involves real-time streaming data processing, machine learning algorithms, and real-time analysis, while traditional data transmission has defects such as high latency and low throughput. In addition, machine learning algorithms rely on manual extraction for feature extraction. There are limitations in its effective feature extraction and feature weighting methods, which limits the classification ability to a certain extent. Therefore, we use 5G technology [32, 33] and deep learning [34, 35] to propose a real-time medical monitoring system for cardiovascular diseases, which is used to process real-time data streams transmitted from wearable devices to predict the health of patients in real time and send timely information to patients.

3 The online prediction system

The real-time online health monitoring system mainly uses open source frameworks such as Kafka, Flink, and Tensorflow to realize the transmission of personal data and the construction of monitoring models. The system can obtain the COVID-19 patients' core information through wearable devices, namely ECG signals, analyze the data with deep learning algorithms in Cloud, and predict the COVID-19 patients' health risk. The overall structure diagram is shown in Fig. 1, which consists of four modules. The first module is responsible for data acquisition and transmission. The second module is responsible for streaming storage and analysis, and the third module is responsible for the training module of deep learning. The fourth stage is the module responsible for health diagnose and predict for Covid-19 patients.

3.1 Details of each module

Acquisition and transmission mainly use acquisition equipment, 5G network infrastructure and other facilities. Details are as follows:

(1) *Acquisition equipment* It is composed of various sensor modules, controllers, processors and power supplies embedded in wearable devices. The main function is to complete the collection, processing and transmission of ECG and Global Positioning System (GPS) positioning data. The selection of these modules needs to meet the needs of usability and wearability, so as to make the wearer feel comfortable and move as much as possible. The photoelectric sensor, ECG sensor and GPS sensor are integrated into the microcontroller. The sensor converts the collected electrical signal into an analog electrical signal, which is processed by the controller and becomes a digital signal, which is transmitted to the handheld device of the individual user through the wireless sensor module. The positioning information and ECG information obtained by the handheld device can be uploaded to the cloud server through the 5G network.

(2) *5G infrastructure* 5g network is a digital cellular network, including supplier service area covered by a series of small geographic area called cellular. The analog signal with the ECG information is digitized on a handheld device, and an analog-to-digital converter turns the digitized information into a

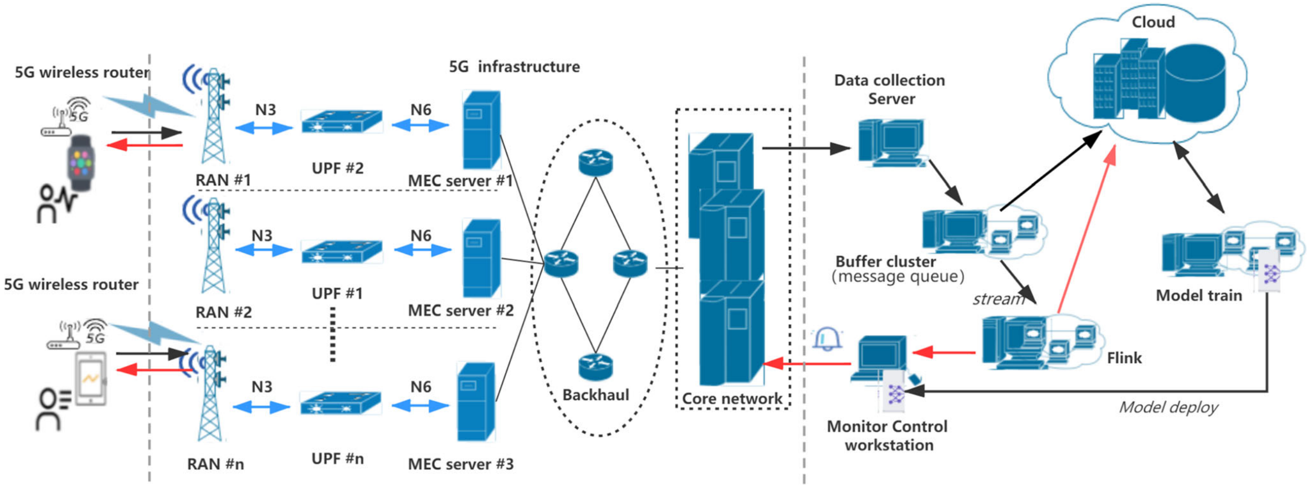

Fig. 1 Architecture design of Cardiovascular Monitoring System for COVID-19 Patients

bitstream for transmission. In the handheld device's network, 5g network local antenna array and automatic sending and receiving devices with low-power consumption and handheld devices to communicate by radio waves [36]. The frequency of the communication channel is assigned by the public pool selection frequency. 5G infrastructure includes 5G access network and 5G core network. Its main purpose is to provide high-bandwidth, low-latency communications. Compared with the traditional network, 5G network has higher access rate and lower delay. It can meet the access requirements of ultra-high traffic density, ultra-high connection number density and ultra-high mobility. It also improves the spectrum efficiency of the network, and reduces the operation and maintenance costs while improving the network energy efficiency. it is very suitable for the real-time system environment of this article.

(3) *Access to 5G infrastructure* Due to the low power consumption, the wearable device adopts a Bluetooth connected to a handheld terminal with 5G connection function (such as a 5G mobile phone), and the data collected by the wearable device are uploaded to the cloud after connecting to the 5G network through the handheld terminal.

Streaming storage and reading stage The framework is mainly divided into two stages, the specific introduction is as described in next section. In the first stage, the message queue collects information from different COVID-19 patients, including the input of source data and the consumption of source data. The second stage is the stream processing pipeline, in which the Flink stream receives the monitoring data stream related to the ECG attribute, and then adopts a batch and stream integrated (streaming for a single COVID-19 patient, and a batch for the entire COVID-19 patients). Perform feature engineering and feature selection operations on the data stream, and wait for data storage and model training.

The training framework of the ECG risk model It is mainly responsible for receiving preprocessed input data, online learning and notification of whether to update our deployed model through the message middleware according to the newly obtained data stream. The initial model of the model is obtained through training by inputting labeled data. Consider the complexity of the model and the continuity of ECG data. We use the CNN+LSTM module as the bottom layer of the model, and the detailed training method is in Section IV. The trained model waits for the monitoring of the monitoring module to achieve the purpose of optimization.

ECG monitoring and risk inference module In order to improve the optimization of the model, we have added a registered monitoring middleware to the monitoring module. The middleware mainly has two functions:

(1) Whether the value of the data in the monitoring system has expired. The time from data entering the system to the model making a risk judgment is much shorter than the processing time after an ECG accident occurs, thereby ensuring the real-time nature of the system.

(2) Another is to monitor whether the model changes. When the model changes, the judgment module is notified to verify whether the model is optimized. So as to achieve the purpose of real-time perfecting the model. Finally, when the model is perfected this time, the output stream is sent back to Kafka again, and the monitoring module will determine whether there is an abnormality in the COVID-19 patients' data stream. For special abnormal information, the alarm function of the system will be triggered. For

normal prediction results, the inference module will push the result predictions to the data center for storage. In this way, through the personal client, COVID-19 patients can obtain personal monitoring information and personalized health reports in real time.

3.2 Details of streaming data

The process of collecting data sets by physical equipment is often real time and continuous. The collected data mainly include COVID-19 patient personal information, ECG information, positioning information, network delay monitoring information, etc. The device processes the data set as time passes, that is to say, it is transmitted to the server receiving end through the handheld terminal in the form of streaming data. The whole service adopts the Client/Server method for individual users, and the receiver accepts data through the Transmission Control Protocol (TCP). However, directly receiving and processing this kind of high-throughput data will cause huge pressure on network resources, so it is necessary to buffer the data when receiving the data. The message queue is used as a buffering solution to effectively solve the problem of data inconsistency. As a high-throughput distributed messaging system, Kafka can support real-time data processing and provide real-time data to the next processor, so Kafka is selected as a data transmission tool.

Secondly, a large number of iterative calculations will be generated in the data processing calculation process, and the processor is required to be able to efficiently support iterative data processing in real time. Choose Flink with higher-level APIs and better benchmark results [37]. Flink provides a wealth of APIs. At the same time, the integration with Druid and Redis is already quite high. We will choose Redis to store the final calculation results. Below we will elaborate on the process of streaming data from the perspective of Kafka and Flink frameworks. The details of streaming data are shown in Fig. 2.

3.2.1 Kafka data stream processing

Input of source data The physical device uses the handheld device to first send to the streaming storage and reading framework via TCP, and then TCP binds the IP and port number to start the data receiving thread. After receiving the data, the system will store the source data in the datasource. In order to effectively analyze and process the original data, the system needs to extract, transform, and load (ETL) a large amount of original data to the target storage data warehouse. In the face of a large number of COVID-19 patient data, a buffer queue is set up from the source data to the data warehouse, namely kafka.

(1) At this stage, Kafka will create topics through Linux commands. Topics are created according to the data types (UserInfo, ECGInfo, GPSInfo). Second, configure the properties of the producer. Finally, various objects are initialized to complete the configuration of Kafka. The configured objects include serialized objects, partitioners, acks objects, etc.

(2) When sending data to the Kafka cluster, in order to achieve Kafka flow card control, the system needs to create a buffer area for the coming data. After setting the buffer data size, when the buffer area is full, it will be sent to Kafka uniformly.

(3) In data parsing, first call the dataParser method, obtain the data type according to the data type position in the data frame definition, and record the data type. At the same time, if the whole frame of data is all 0 during the analysis, it means that the frame data is invalid data, and the frame data is skipped. After analyzing the data type, enter the analysis function of different data items according to the data type, analyze the whole data, and store it in the form of an object.

(4) When each test item creates the ProducerRecord object, the source data needs to be reorganized to form a processing form in Flink. There are two objects involved in sending messages, KafkaProducer and ProducerRecord. ProducerRecord specifies the topic information that needs to be sent, the message content value, and can also specify partition information and key values. After the partition is selected, the producer can determine which topic and which partition to send the message to. Inside the Producer, a separate thread will send the record to the corresponding broker.

(5) After the Kafka server successfully receives the message sent by the Producer, it will respond with a response. If the message is written successfully, it will return a RecordMetaData object. If it fails, the Producer will resend the message. If it fails after a few times, it will return an error message.

Consumption of source data After Kafka processes the data, consumers need to subscribe to the message and read the data in Kafka. In this article, Kafka consumers are the target storage data warehouse and the Flink clusters. In Flink, Kafka's Partition needs to correspond to Flink's parallel task instance. Flink can also guarantee that even after a failure, the allocation of partitions to Flink instances can be maintained, so partition determinism is maintained, which ensures that data processing is exactly at once. This effectively guarantees the integrity of the system data.

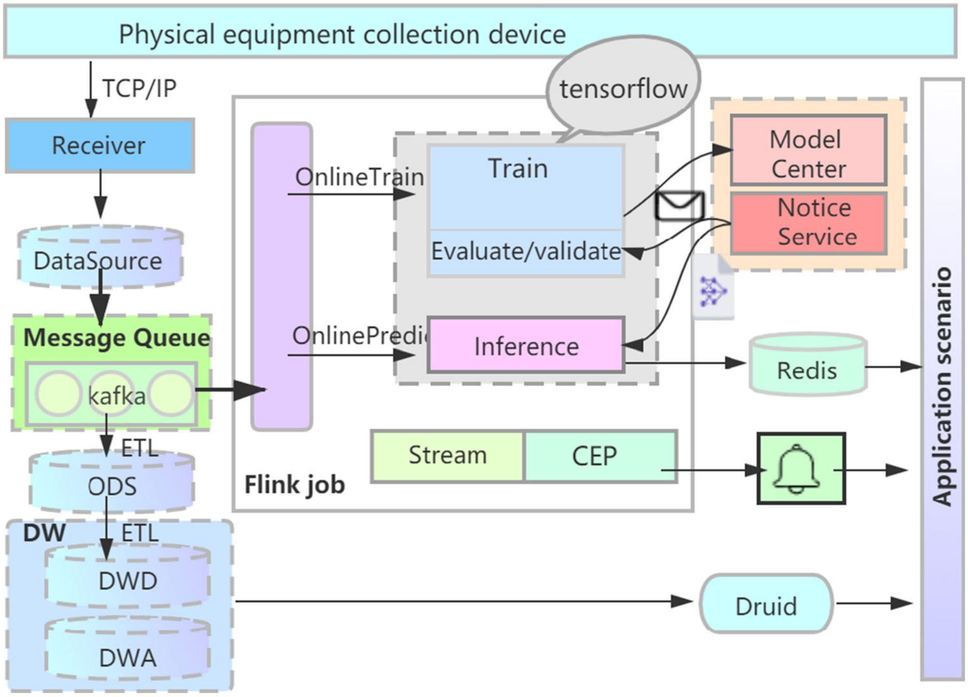

Fig. 2 ECG data flow process

3.2.2 Flink data stream processing

After confirming that the Kafka consumer is Flink, it is necessary to create a Flink-Kafka consumer object. The realization of a data processing process based on Flink mainly includes the following five steps. First, you need to obtain the execution environment for data processing, and then load the initial data. After the data are loaded into Flink, specify the data conversion method, that is, to realize the specific data processing process, that is, the specific deep learning-based classification method in Sect. 4. In addition, the model update is also based on the data collection situation within a period of time to update the model, the notice service will let the new model act on the system's prediction. After the processing is completed, specify the storage location of the data calculation result. Since Flink is lazy loaded, after the above steps are defined, the execution of the program needs to be triggered at the end before the Fink cluster will start processing data. For patients detected as suspected of COVID-19 patient, the system will also alert management personnel to deal with such situations.

4 Cardiovascular disease classification algorithm based on Deep Learning

4.1 Data preprocessing

The ECG signal is a weak electrical signal, which is easily interfered by electrical signals from various other sources, including baseline drift, EMG interference and power frequency interference. In order to obtain a truly useful ECG signal, we need to denoise these high-frequency or low-frequency noises. In addition, a complete ECG is composed of a long continuous time series. If it is directly input into the neural network classification, it will greatly increase the complexity of the network calculation, and it is not conducive to extracting good feature information. Therefore, in the study of ECG signal classification, the entire ECG signal is usually divided into several small segments in units of heart beats according to specific rules. The denoising of ECG signals and heartbeat segmentation are collectively referred to as preprocessing.

4.1.1 ECG signal denoising

The ECG signal contains a variety of different types of high-frequency or low-frequency noise. The purpose of denoising the ECG signal is to suppress the noise in the signal, and to enhance the part that can contribute to feature extraction. In order to obtain useful signals, this paper uses discrete wavelet transform (DWT) to process ECG signals. Because the tightness of the ECG signal processing has a greater impact on the signal, *Daubechies* wavelet is more suitable. In this paper, DB8 wavelet basis is selected to decompose the ECG signal into 8 layers, and the coefficients of each layer are obtained. The frequency difference of the three noises of ECG signal is relatively large, so the soft threshold processing method is selected, that is, different thresholds are used for quantization processing at different transform scales. Finally, the ECG signal is reconstructed according to the low-frequency coefficients of the 8*th* layer obtained by DB8 wavelet decomposition and the high-frequency coefficients of each layer, and the denoised ECG signal is obtained. The evaluation method of the denoising effect of the ECG signal is reflected by the signal-to-noise ratio (SNR) and the mean square error (MSE). The larger the SNR value, the less noise the denoised ECG signal contains and the better the denoising effect; the smaller the MSE, the smaller the degree of distortion of the ECG signal. The calculation method is shown in formula (1) and (2) where $s(i)$ represents the original ECG signal containing noise, $x(i)$ represents the denoised ECG signal, and N represents the length of the collected ECG signal.

$$SNR = 10 \cdot \lg \sum\nolimits_{\mathrm{n}=1}^{N} \frac{s(i)^2}{[x(i) - s(i)]^2} \tag{1}$$

$$MSE = \frac{\sum_{n=1}^{N} [x(i) - s(i)]^2}{N} \tag{2}$$

4.1.2 Heartbeat segmentation

After the ECG signal is denoised, the R peak in the waveform needs to be located and segmented. For the R peak detection task, combined with the real-time requirements of ECG signal detection, the current mainstream R peak positioning method P-T algorithm is used [38]. According to the characteristics of ECG signals, the algorithm uses the characteristics of large slope of QRS complexes, and searches for the peak value of R waves in ECG signals through differentiation and adaptive threshold methods to achieve positioning effect. After the detection of the R peak position is completed, the heart beat segmentation of the entire ECG signal is performed, and the R peak is taken as the reference position, and the forward and backward intercepts are performed, respectively. The intercepted length is at least a complete heartbeat, and a complete heartbeat is about 0.6s ~ 0.8s, so the sampling point must be greater than 360*0.8 = 288. Too many sampling points will cause correlation interference between different types of waveforms. This paper selects 300 sampling points, divides all the ECG signals in MIT-BIH, and normalizes each collected heartbeat, limiting the amplitude of each heartbeat to $[-1, 1]$.

4.1.3 Data enhancement

ECG classification is an imbalance problem. Abnormal heartbeats are much smaller than normal heartbeats. Because deep learning has a strong ability to express and explain, it is difficult for the model to learn a small number of sample features during the training process, which makes the model invalid in practical applications. Therefore, less data must be enhanced to solve the imbalance problem. This paper adopts the Synthetic Minority Oversample Technique (SMOTE) oversampling algorithm proposed by Chawla in 2002 [39]. The idea of this algorithm is to synthesize new minority samples through a certain strategy. The synthesis strategy is for each minority sample *a*, A sample *b* is randomly selected from its nearest neighbor, and then a point on the line between *a* and *b* is randomly selected as the newly synthesized minority sample. Therefore, this paper uses the algorithm to synthesize new sample data from the similarities between the existing minority heartbeat samples. The SMOTE algorithm is shown in formula (3).

$$X_{new} = X_i + \mathrm{rand}(0, 1) * (X_{i,j} - X_j) \tag{3}$$

where rand(0, 1) represents the random number generated between 0 and 1, X_{new} represents the newly generated heartbeat samples of S, V, F and Q, X_i represents the *i*th heartbeat sample in the minority class, $X_{i,j}$ represents the heartbeat sample *b* in the *i*th neighborhood of the *i*th heartbeat sample *a* in the minority class.

4.2 Convolutional neural networks

CNN is a kind of multilayer neural network used for image classification, segmentation or detection that has developed rapidly in recent years [40, 41]. It inherits the unique feature extraction capabilities of deep learning. Because of its advantages such as local connection, weight sharing, and down-sampling, it effectively reduces the number of parameters in the neural network structure with a large amount of data, reduces the complexity of the operation, and reduces the memory of the operation. CNN is constructed by different combinations of input layer, hidden layer and output layer. The hidden layer usually includes a convolutional computing layer, a pooling computing layer, and a fully connected layer. The network structure of 1D-CNN is shown in Fig. 3.

The convolution operation layer is to perform convolution operation on the input sequence or picture, and the purpose is to extract the characteristics of the input signal. Convolution operation includes convolution operation and activation function. The convolution operation is to multiply a set of weights with the input, expressed in matrix form as formula (4), where X represents the matrix

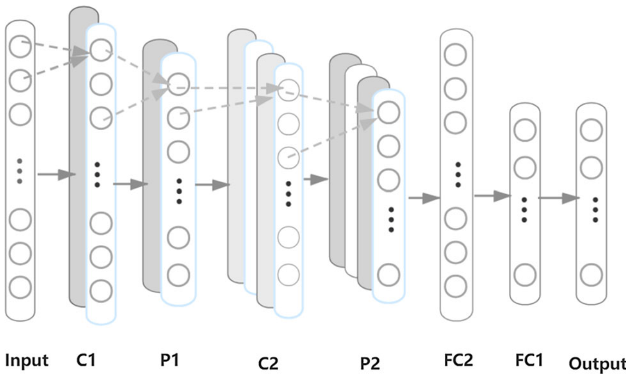

Fig. 3 1D-CNN Network Structure

representation of the input signal and W is the convolution kernel. The size and number of the weight matrix can be customized by experience; $*$ means convolution. After the convolution operation, the result needs to be determined and transformed by the activation function, and the output that reaches the threshold is mapped to another space through nonlinear changes, and then the features can be nonlinearly classified.

$$S(t) = (X * W)(t) \tag{4}$$

The pooling operation layer is to down-sample the features extracted by the convolutional layer. This process can retain important pair of feature information without changing the number of feature maps. In this way, the model reduces spatial information to obtain better computing performance and reduces the risk of model overfitting.

The final classifier in the entire CNN network is implemented in the fully connected layer, and the output after convolution and pooling is flattened into a single value vector, and the probability calculation is performed through the softmax function to obtain the final category output. The softmax function converts the input into a probability value ranging from 0 to 1, and the sum of all probability values is 1.

This paper uses a network composed of multiple convolutional layers and pooling layers to extract features of the ECG signal. Feature extraction technology can replace manual labor, avoiding the inaccuracy of feature selection due to human reasons and saving a lot of time spent on features.

4.3 Long short-term memory networks

Long short-term memory network (LSTM) is a variant of recurrent neural network. The traditional RNN hidden layer is used as a memory unit. As the model progresses over time, the effective information of the input data is gradually weakened. The LSTM redesigned the memory module to retain the backpropagation error between the time step and the level, so that the network model continues to maintain the learning state in multiple time steps and thus has the ability to capture the causality of long-distance information. Therefore, LSTM is suitable for ECG signals with timing characteristics.

The core of LSTM is composed of input gate, output gate and forget gate. These three control gates can enable LSTM neurons to read, write, reset and update long-distance historical information. The structure diagram is shown in Fig. 4. The overall calculation formula is as follows.

LSTM realizes the selective loss of information in neurons through the forget gate in the structure, and the calculation formula is as (5).

$$\mathrm{f}_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{5}$$

where W_f represents the weight matrix of the forgetting gate, h_{t-1} is the previous output in the network, x_t represents the current input, b_f represents the bias term of the forgetting gate, and $\sigma(\cdot)$ represents the sigmoid function.

The output of the input gate consists of two parts. The calculation formulas are as (6) and (7). The output of the input gate is composed of two parts, i_t represents the current output, which is realized by the sigmoid function, and $\tilde{C}_t$ represents the current state, which is realized by the tanh function.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{6}$$

$$\tilde{C}_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{7}$$

Among them, i_t represents the current output, realized by the sigmoid function, $\tilde{C}_t$t represents the current state, realized by the tanh function, $W_{(\cdot)}$ represents the weight matrix of this part, and $b_.$ represents the bias term of this part.

The output gate calculation formulas are as (8), (9) and (10).

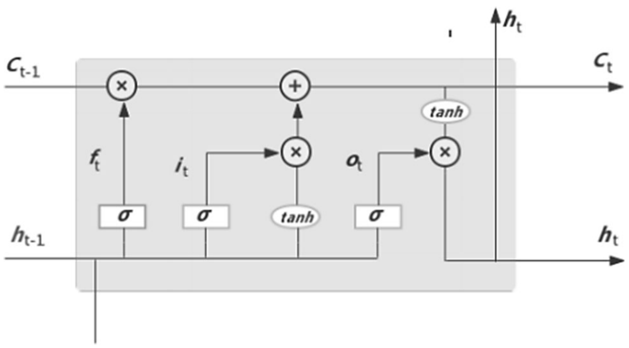

Fig. 4 LSTM Network Structure

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{8}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{9}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{10}$$

C_t is the output state of the cell, which is composed of the product of the output f_t of the forgetting gate and the state C_{t-1} at the previous moment and the sum of the product of the two outputs in the input gate. W_o represents the weight matrix of the output gate, and b_o represents the offset term of the output gate, h_t represents the current final output.

In LSTM, the current state information c_t and the previous state information c_{t-1} have a linear relationship. When the forget gate is open, that is, when the output of the sigmoid unit is close to 1, the gradient will not disappear, and the new state The information is the weighted average of the previous state information and the accumulated information at the current moment, so regardless of the sequence length, as long as the forget gate is open, the network can remember the past state information, that is, LSTM can capture long-term dependencies.

4.4 ECG signal classification based on CNN+LSTM

Both CNN and RNN can classify image and text input, so there is an opportunity to combine the two network models to improve classification efficiency. If the input adds time characteristics that CNN itself cannot handle, the combination of the two is more advantageous. Since the ECG signal is a physiological signal collected in accordance with time, it contains rich time domain features. The positional relationship between the waveforms of various stages in a heartbeat beat is close. The input of the neuron in LSTM is not only affected by the input at the current moment, but also related to the output at the previous moment, that is, there is an association between nodes at different moments in the time series, which can save contextual information, and is dependent on long-distance time. Time series are particularly effective. Therefore, this paper uses the local perceptual field characteristics of CNN and the memory function of LSTM to construct a classification model combining CNN+LSTM. The specific model structure is shown in Fig. 5. Among them, the size of the convolution kernel is increased from 21 by 2, and the sliding step size is 2 to extract the morphological characteristics of the ECG signal. The number of convolution kernels is 4, 16, 32, and 64. The convolutional layer performs feature reorganization to form a feature map. The pooling part selects a maximum pooling operation with a size of 3 and a step size of 2 to compress the feature vector size. The signal features extracted by the convolutional network are sent to the 128-unit LSTM network for time analysis. Finally, various predicted probabilities are output through the fully connected layer and the softmax function.

Model training process

- **CNN extract features** Input the preprocessed data into the CNN network, and extract intermediate features after convolution and pooling. The calculation formula is as (11).

$$X_j^n = Relu(\sum_{j \in M_j}(X_j^{n-1} * W_j^n) + b_j^n) \tag{11}$$

  Among them, X_j^n represents the jth feature of the ECG signal after the nth layer of convolution, W represents the convolution kernel, and b represents the bias term.
- **LSTM extraction features** take the middle feature X_j^n of the ECG signal after feature extraction as the input of the LSTM layer, and use the formula introduced in the previous section to calculate the output.
- **Softmax classification** The signal features extracted by CNN and LSTM are sent to the fully connected layer, where 5 classification labels are encoded with one-hot, and then the softmax function is used to generate the probability p_k of each heartbeat type. The calculation formula is as (12).

$$p_k = softmax(x)\frac{exp(\theta_k^T x)}{\sum_k exp(\theta_k^T x)} \tag{12}$$

  where x is the input sample data, k is the heartbeat type, and θ is the model parameter. In this paper, x represents each heartbeat, k = 1,2,3,4,5, corresponding to categories N, S, V, F and Q, respectively.
- **Backpropagation and weight update**: After the category is judged, the ECG classification loss is calculated, and the loss is back-propagated according to the chain rule to calculate the gradient of each weight and use gradient descent to update the weights. The calculation formula is as (13).

$$\theta_j = \theta_j - \alpha\frac{\nabla J(\theta)}{\nabla \theta_j}, \quad j = 1, 2, \ldots, k \tag{13}$$

- **Iterative training** Repeat the above steps until the network converges or reaches the maximum number of training cycles. If the effect of the model training result is improved, save the model.

Model testing process

Load the optimal model. Load the optimal parameters of the model training stage, input the ECG signal data of the test set into the CNN+LSTM network for calculation, and output the final results and evaluation indicators.

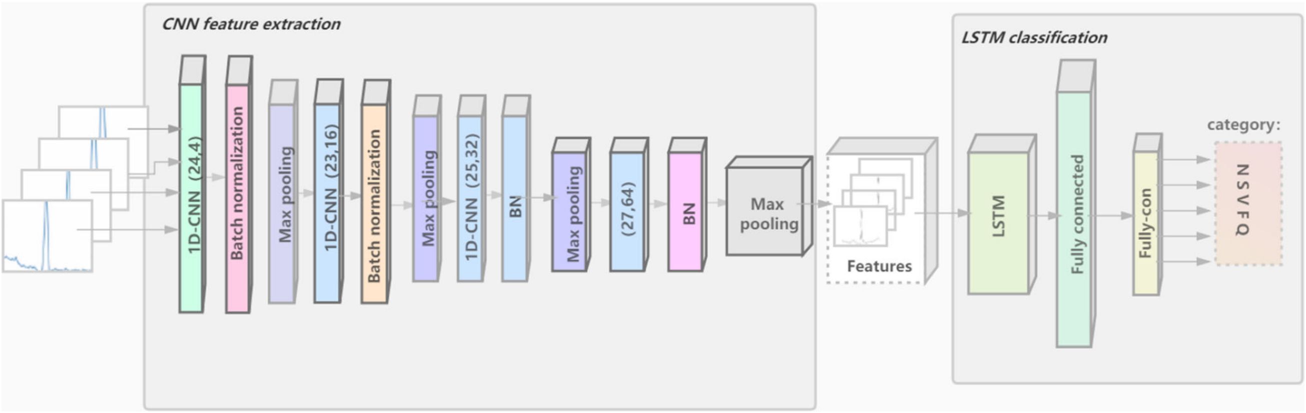

Fig. 5 CNN-LSTM Network Structure

4.5 Discussion and analysis

In the patient's ECG signal, usually abnormal ECG signal data are far less than normal ECG data. This unbalanced data distribution often leads to models that are more inclined to learn with multiple data categories, and the learning of a few categories of data is insufficient, especially in the process of small-batch gradient descent optimization. If there are only a few abnormal ECG signal data in the ECG data set, it makes the direction of gradient descent heavily dependent on normal ECG data, resulting in a low recognition rate of abnormal ECG signals. Usually, the solution to this problem is to perform multi-sampling of categories with a small amount of data or under-sampling categories with a large amount of data. In recent years, many people have used the popular generative confrontation network in deep learning to enhance data to achieve data balance.

5 Experimental results and discussion

5.1 Dataset introduction

This paper uses the most widely used ECG signal database in the field of ECG signal classification research, namely the MIT-BIH arrhythmia database. The MIT-BIH ECG database contains 48 half-hour records of the 24-hour dual-channel ECG records of 47 subjects. Among them, 23 records of the 100 series are randomly selected from more than 4000 Holter collectors, and the other 25 records of the 200 series are unusual but clinically important arrhythmia signals. The ECG signal data are stored in a binary format with a sampling frequency of 360 Hz, and the atr file in each record indicates the type of heartbeat. This paper is classified into 5 categories according to AAMI standards, namely N (normal heart beat), S (supraventricular odor), V (ventricular odor), F (fusion heart beat), Q (undefined heart beat). The number of various heartbeats is shown in Table 1.

5.2 Experimental settings and evaluation indicators

The hardware configuration of the experimental platform is Intel i7-6700 CPU, the graphics card is GTX1080Ti, the memory is 32G, the operating system is Window10 system, and the model is implemented based on the Python programming language and Tensorflow framework. When training the network model parameters, the initial value of the learning rate is 0.001, and then the ReduceLROnPlateau function makes the learning rate adaptive to the model; the Dropout parameter is 0.2, Batch_size is 128, and Epoch is 1000. For S, V, F, and Q, first select 20% as the test set, and use the remaining 80% to generate new data. In order to maintain the credibility of the experiment, this paper adopts a tenfold cross-validation method.

In order to evaluate the performance of the model in this paper, the following two indicators are used:

(1) In medical diagnosis, normal heartbeats and abnormal heartbeats are negative and positive respectively. If the true type is negative and classified as negative, it is recorded as true negative (TN); the true type is positive and classified as negative and recorded as false negative (FN); the true type is classified as negative. Negatives are classified as positives and counted as false positives; those that are true to positives are classified as negatives and counted as true positives (TP). In the ECG signal judgment model, Accuracy (Acc), Specificity (Spe) and Sensitivity (Sen) are generally used as indicators for judgment. Among them, accuracy represents the probability that the model is accurately classified for a given test set, and specificity represents the probability that a negative example is accurately predicted by the model, and sensitivity represents the probability that a positive example is correctly

Table 1 Number of various heartbeats

Heartbeat type	Description	Number
N (Normal heart beat)	Normal beat	83513
	Left bundle branch block beat	
	Right bundle branch block beat	
	Atrial escape beats	
	Nodal(junctional) escape	
S (Supraventricular odor)	Atrial premature	2184
	Aberrant atrial premature	
	Nodal(junctional) premature	
	Supra-Ventricular premature	
V (Ventricular odor)	Premature Ventricular	6975
	contraction Ventricular escape	
F (Fusion heart beat)	Fusion of Ventricular and Normal	801
Q (Undefined heart beat)	Paced	3593
	Fusion of Paced and Normal	
	Unclassifiable	
Total		97066

classified by the model. The three evaluation index formulas are as (14), (15) and (16).

$$Sen = \frac{TP}{TP + TN} \times 100\% \tag{14}$$

$$Spe = \frac{TN}{TN + FP} \times 100\% \tag{15}$$

$$Acc = \frac{TN + TP}{TP + FP + FN + TN} \times 100\% \tag{16}$$

(2) ROC curve: This indicator is based on the three models of SVM, CNN, LSTM and the scheme of this article. After training 4 models, they are used to predict the test set and obtain the ROC curve. The area under the ROC curve can be used to judge the performance of the model.

5.3 Result analysis

In (1), the probability of Acc, Spe and Sen in different methods of cardiovascular disease are shown in Fig. 6. As can be seen from the this figure, through the predictions of different models on the test set, it is found that the classification effect of traditional machine learning SVM is less than that of deep learning methods, and our proposed method is higher than other deep learning methods in Acc, Spe, and Sen. They reached 99.29%, 99.53%, and 97.77%, respectively.

In (2), four trained models are used to predict the test set, and the ROC curve is shown in Fig. 7. It can clearly see that the ROC curves of these four models have a large deviation from the 45-degree diagonal. Among them, the area under the SVM model is small, and the trained SVM model has a general classification effect on the ECG signals of coronary patients. This shows that traditional machine learning has a large amount of data and has a poor

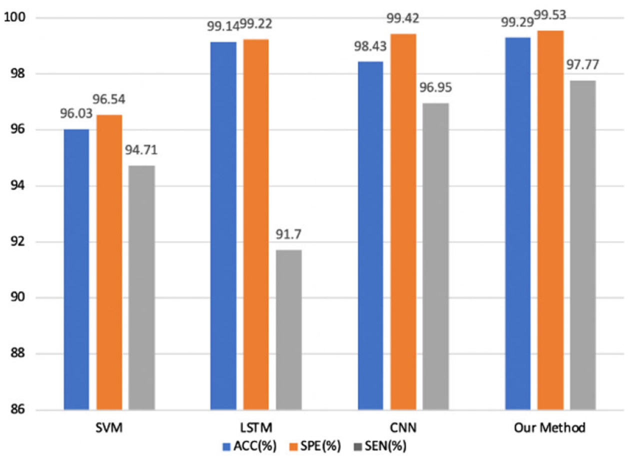

Fig. 6 Acc, Spe and Sen for each model

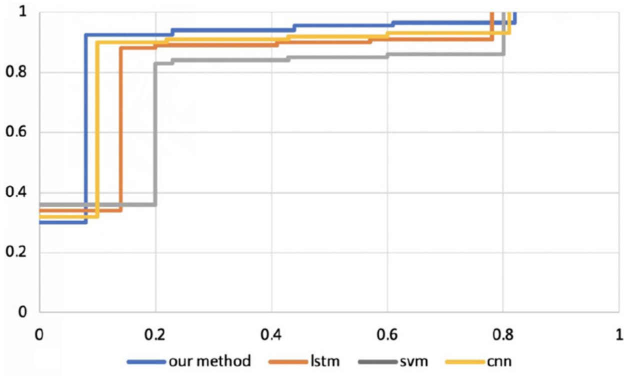

Fig. 7 ROC curve comparison chart

feature extraction effect on data with more features, which makes the generalization of the model poor; the area under CNN and LSTM is not much different, and the ECG signal classification effect is good. It also shows that the deep learning model can effectively extract the unilateral features of the ECG data to a certain extent, making the model generalization better; it can be seen that the area of our proposed method is the largest, showing that the CNN+LSTM models can not only propose the characteristics of the ECG signal itself, but also accurately propose the time-domain features in it. Combining them for classification can effectively improve the classification. The effect makes the model strong in generalization.

6 Conclusion

Aiming at the accuracy and timeliness of heart rate detection for COVID-19 patients, this paper proposes a real-time cardiovascular monitoring system based on 5G and deep learning to ensure low latency and high throughput of ECG signal data transmission in wearable devices. In order to achieve real-time ECG signal data monitoring, analysis and diagnosis, it adopts the new generation of wireless communication technology 5G, plus the real-time data processing platform Flink framework, and finally uses the classification of the proposed deep learning ECG signal model to realize real-time ECG signal data monitoring, analysis and diagnosis. The correct rate of model prediction can reach 99.29%, which proves that the real-time medical monitoring system is necessary and can effectively predict the actual situation of the COVID-19 patient's cardiovascular system. It has high practical value and can realize early warning of emergencies in time.

Our next plan is to further optimize the deep learning model, try to increase the one-dimensional time series signal to the two-dimensional space, use the Generative Adversarial Networks to enhance the data of the insufficient number of heartbeat types, solve the problem of ECG signal imbalance, and consider using migration learn to increase model accuracy and reduce training time.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grant No. 61373162, in part by the Sichuan Provincial Science and Technology Department Project under Grant No. 2019YFG0183, in part by the Sichuan Provincial Key Laboratory Project under Grant No. KJ201402, and in part by the Japan Society for the Promotion of Science (JSPS) Grants-in-Aid for Scientific Research (KAKENHI) under Grant JP18K18044 and JP21K17736.

Declarations

Conflicts of interest The authors declare that there is no conflict of interest regarding the publication of this article.

References

1. Zhou X, Hu Y, Liang W, Ma J, Jin Q (2020) Variational lstm enhanced anomaly detection for industrial big data. IEEE Trans Ind Inform 17(5):3469-3477. https://doi.org/10.1109/TII.2020.3022432
2. Yu K, Tan L, Shang X, Huang J, Srivastava G, Chatterjee P (2021) Efficient and privacy-preserving medical research support platform against covid-19: a blockchain-based approach. IEEE Consumer Electron Mag 10(2):111–120
3. Yu K, Tan L, Aloqaily M, Yang H, Jararweh Y (2021) Blockchain-enhanced data sharing with traceable and direct revocation in iiot. IEEE Trans Ind Inf. https://doi.org/10.1109/TII.2021.3049141
4. Yu K, Guo Z, Shen Y, Wang W, Lin JC-W, Sato T (2021) Secure artificial intelligence of things for implicit group recommendations. IEEE Internet Things J. https://doi.org/10.1109/JIOT.2021.3079574
5. Manogaran G, Lopez D (2018) Health data analytics using scalable logistic regression with stochastic gradient descent. Int J Adv Intell Paradigms 10(1–2):118–132
6. Zhou X, Liang W, Wang KI, Wang H, Yang LT, Jin Q (2020) Deep-learning-enhanced human activity recognition for internet of healthcare things. IEEE Int Things J 7(7):6429–6438
7. Wang J, Huang X, Tang S, Shi GM, Ma X, Guo J (2019) Blood triglyceride monitoring with smartphone as electrochemical analyzer for cardiovascular disease prevention. IEEE J Biomed Health Inf 23(1):66–71
8. Li MH, Yadollahi A, Taati B (2017) Noncontact vision-based cardiopulmonary monitoring in different sleeping positions. IEEE J Biomed Health Inf 21(5):1367–1375
9. Saadatnejad S, Oveisi M, Hashemi M (2020) Lstm-based ecg classification for continuous monitoring on personal wearable devices. IEEE J Biomed Health Inf 24(2):515–523
10. Hu H, Wen Y, Chua T-S, Li X (2014) Toward scalable systems for big data analytics: a technology tutorial. IEEE Access 2:652–687
11. Zhang X, Yang L, Ding Z, Song J, Zhai Y, Zhang D (2021) Sparse vector coding-based multi-carrier noma for in-home health networks. IEEE J Select Areas Commun 39(2):325–337. https://doi.org/10.1109/JSAC.2020.3020679
12. Feng C, Yu K, Bashir AK, Al-Otaibi YD, Lu Y, Chen S, Zhang D (2021) Efficient and secure data sharing for 5g flying drones: a blockchain-enabled approach. IEEE Netw 35(1):130–137
13. Li S, DaXu L, Zhao S (2015) The internet of things: a survey. Inf Syst Front 17(2):243–259
14. Yu K, Tan L, Lin L, Chen X, Yi Z, Sato T (2021) Deep learning empowered breast cancer auxiliary diagnosis for 5gb remote e-health. IEEE Wireless Commun. https://doi.org/10.1109/MWC.001.2000374
15. Sannino G, DePietro G (2018) A deep learning approach for ecg-based heartbeat classification for arrhythmia detection. Future Generat Comput Syst 86:446–455
16. Ed-daoudy A, Maalmi K (2019) A new internet of things architecture for real-time prediction of various diseases using machine learning on big data environment. J Big Data 6(1): 104
17. Sun J, Reddy C.K. (2013) "Big data analytics for healthcare," In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 1525–1525
18. Nair LR, Shetty SD, Shetty SD (2018) Applying spark based machine learning model on streaming big data for health status prediction. Comput Electr Eng 65:393–399
19. Alotaibi S, Mehmood R, Katib I, Rana O, Albeshri A (2020) Sehaa: a big data analytics tool for healthcare symptoms and

diseases detection using twitter, apache spark, and machine learning. Appl Sci 10(4):1398

20. Ed-Daoudy A, Maalmi K (2019) “Real-time machine learning for early detection of heart disease using big data approach. In: 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS). IEEE, pp. 1–5
21. Manogaran G, Varatharajan R, Priyan M (2018) Hybrid recommendation system for heart disease diagnosis based on multiple kernel learning with adaptive neuro-fuzzy inference system. Multimed Tools Appl 77(4):4379–4399
22. Melin P, Miramontes I, Prado-Arechiga G (2018) A hybrid model based on modular neural networks and fuzzy systems for classification of blood pressure and hypertension risk diagnosis. Expert Syst Appl 107:146–164
23. Al-Makhadmeh Z, Tolba A (2019) Utilizing iot wearable medical device for heart disease prediction using higher order boltzmann model: a classification approach. Measurement 147:106815
24. Yin H, Jha NK (2017) A health decision support system for disease diagnosis based on wearable medical sensors and machine learning ensembles. IEEE Trans Multi-Scale Comput Syst 3(4):228–241
25. Jabeen F, Maqsood M, Ghazanfar MA, Aadil F, Khan S, Khan MF, Mehmood I (2019) An iot based efficient hybrid recommender system for cardiovascular disease. Peer-to-Peer Netw Appl 12(5):1263–1276
26. Kumar PM, Lokesh S, Varatharajan R, Babu GC, Parthasarathy P (2018) Cloud and iot based disease prediction and diagnosis system for healthcare using fuzzy neural classifier. Future Generat Comput Syst 86:527–534
27. Kumar PM, Gandhi UD (2018) A novel three-tier internet of things architecture with machine learning algorithm for early detection of heart diseases. Comput Electr Eng 65:222–235
28. Hammad M, Alkinani MH, Gupta BB et al (2021) Myocardial infarction detection based on deep neural network on imbalanced data. Multimed Syst. https://doi.org/10.1007/s00530-020-00728-8
29. Sedik A, Hammad M, Abd El-Samie FE et al (2021) Efficient deep learning approach for augmented detection of Coronavirus disease. Neural Comput Appl. https://doi.org/10.1007/s00521-020-05410-8
30. AlZu’bi S, Shehab M, Al-Ayyoub M, Jararweh Y, Gupta B (2020) Parallel implementation for 3d medical volume fuzzy segmentation. Patt Recognit Lett 130:312–318
31. Abd El-Latif A.A, Abd-El-Atty B, Hossain M.S, Rahman M.A, Alamri A, Gupta B.B. (2018) Efficient quantum information hiding for remote medical image sharing. IEEE Access 6: 21 075–21 083
32. Zhen L, Bashir AK, Yu K, Al-Otaibi YD, Foh CH, Xiao P (2021) Energy-efficient random access for leo satellite-assisted 6g internet of remote things. IEEE Int Things J 8(7):5114–5128
33. Tan L, Xiao H, Yu K, Aloqaily M, Jararweh Y (2021) A blockchain-empowered crowdsourcing system for 5g-enabled smart cities. Comput Standards Interfaces 76:103517
34. Zhang J, Yu K, Wen Z, Qi X, Paul AK (2021) 3d reconstruction for motion blurred images using deep learning-based intelligent systems. Comput Mater Continua 66(2):2087–2104
35. Yu K, Lin L, Alazab M, Tan L, Gu B (2020) Deep learning-based traffic safety solution for a mixture of autonomous and manual vehicles in a 5g-enabled intelligent transportation system. IEEE Trans Intell Transp Syst. https://doi.org/10.1109/TITS.2020.3042504
36. Gong Y, Zhang L, Liu RP, Yu K, Srivastava G (2021) Nonlinear mimo for industrial internet of things in cyber-physical systems. IEEE Trans Ind Inf 17(8):5533–5541. https://doi.org/10.1109/TII.2020.3024631
37. Greco L, Ritrovato P, Xhafa F (2019) An edge-stream computing infrastructure for real-time analysis of wearable sensors data. Future Generat Comput Syst 93:515–528
38. Pan J, Tompkins WJ (1985) A real-time qrs detection algorithm. IEEE Trans biomed Eng 3:230–236
39. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: synthetic minority over-sampling technique. J Artif Intell Res 16:321–357
40. Guo Z, Yu K, Jolfaei A, Bashir AK, Almagrabi AO, Kumar N (2021) A fuzzy detection system for rumors through explainable adaptive learning. IEEE Trans Fuzzy Syst. https://doi.org/10.1109/TFUZZ.2021.3052109
41. Guo Z, Shen Y, Bashir AK, Imran M, Kumar N, Zhang D, Yu K (2021) Robust spammer detection using collaborative neural network in internet of thing applications. IEEE Int Things J 8(12):9549–9558. https://doi.org/10.1109/JIOT.2020.3003802

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Liang Tan[1,2] · Keping Yu[3] · Ali Kashif Bashir[4,5] · Xiaofan Cheng[1] · Fangpeng Ming[1] · Liang Zhao[6] · Xiaokang Zhou[7]

✉ Keping Yu
keping.yu@aoni.waseda.jp

Liang Tan
jkxy_tl@sicnu.edu.cn

Ali Kashif Bashir
Dr.alikashif.b@ieee.org

Xiaofan Cheng
978029808@qq.com

Fangpeng Ming
11451287@qq.com

Liang Zhao
lzhao@sau.edu.cn

Xiaokang Zhou
zhou@biwako.shiga-u.ac.jp

[1] College of Computer Science, Sichuan Normal University, Chengdu 610101, China

[2] China and Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China

[3] Global Information and Telecommunication Institute, Waseda University, Tokyo, Japan

[4] Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, UK

[5] School of Information and Communication Engineering, University of Electronics Science and Technology of China (UESTC), Chengdu, China

[6] School of Computer Science, Shenyang Aerospace University, Shenyang 110136, China

[7] Faculty of Data Science, Shiga University, Hikone, and RIKEN Center for Advanced Intelligence Project, Tokyo, Japan