Pomiary Automatyka Robotyka, ISSN 1427-9126, R. 28, Nr 2/2024, 21–36, DOI: 10.14313/PAR_252/21

The ECG Signal Monitoring System Using Machine Learning Methods and LoRa Technology

Sandra Śmigiel, Tomasz Topoliński
Bydgoszcz University of Science and Technology, Faculty of Mechanical Engineering, al. prof. S. Kaliskiego 7, 85-796 Bydgoszcz

Damian Ledziński, Tomasz Andrysiak
Bydgoszcz University of Science and Technology, Faculty of Telecommunications, Computer Science and Electrical Engineering, al. prof. S. Kaliskiego 7, 85-796 Bydgoszcz

Abstract: An electrocardiogram (ECG) is the first step in diagnosing heart disease. Heart rhythm abnormalities are among the early signs of heart disease, which can contribute to a patient's heart attack, stroke, or sudden death. The importance of the ECGs has increased with the development of technologies based on machine learning and remote monitoring of vital signs. In particular, early detection of arrhythmias is of great importance when it comes to diagnosing a patient with heart disease. This is made possible through recognizing and classifying pathological patterns in the ECG signal. This paper presents a system for mobile monitoring of ECG signals enriched with the results of the study of the application of machine learning models from the group of Tree-based ML techniques and Neural Networks in the context of heart disease classification. The research was carried out through the use of the publicly available PTB-XL database of the ECG signals. The results were analyzed in the context of classification accuracy for 2, 5 and 15 classes of heart disease. Moreover, a novelty in the work is the proposal of machine learning techniques and architectures neural networks, which, have been selected to be applicable to IoT devices. It has been proven that the proposed solution can run in real time on IoT devices.

Keywords: mobile device, ECG signal, classification, IoT, LoRa, Neural Network, Machine Learning, PTB-XL database

1. Introduction

Cardiology is one of the fields of medicine where devices are finding applications in both diagnosis and therapy. Basic and relatively simple to obtain biomedical signals are increasingly providing useful and even relevant information about a patient's condition (e.g., blood pressure, heart rate, heart rhythm). When it became possible to remotely transmit information recorded by medical devices, concepts of telemonitoring and even remote treatment began to emerge.

Remote monitoring in cardiology involves the use of electronic devices and telecommunications technology to digitally transmit physiological data from a patient's home to a healthcare facility, thus helping to monitor the progression of diseases. This can be partially achieved using the parameters mentioned earlier, such as blood pressure and heart rate, which are measured by devices such as blood pressure monitors and pulse oximeters. In contrast, a comprehensive approach is the use of electrocardiography (ECG). ECG, as a graphical representation of the heart's electrical activity, makes it possible to detect and analyze abnormal electrical activity of the heart outside the hospital, during daily activities. ECG telemonitoring is responsible for reading the recorded ECG signals, archiving them on a mobile device (saving) and transmitting them to a medical center (sending).

The electrocardiogram (ECG) is determined by many variables. It provides specific curves of electrical differences when the atria and ventricles of the heart muscle depolarize and repolarize, which corresponds to the cardiac cycle. The work of the atria and ventricles of the heart muscle produces waves as a result. The ECG signal consists of repetitive waves, such as the P-wave, QRS complex, and T-wave. These elements make up the ECG signal (fig. 1). The ECG analysis is one of the most important steps in diagnosing a heart disease. Issues in this area involve finding pathological patterns in the ECG signal. Any abnormality in the heart muscle can cause a heart disease, which may contribute to heart attacks, strokes, and cardiac arrhythmias. Many of these are treatable if they are diagnosed in time and the treatment undertaken by doctors is properly conducted. Predicting the onset of these disorders has been a major concern of researchers and scientists for many years [2].

Autor korespondujący:
Tomasz Andrysiak, tomasz.andrysiak@pbs.edu.pl

Artykuł recenzowany
nadesłany 30.11.2023 r., przyjęty do druku 15.05.2024 r.

Zezwala się na korzystanie z artykułu na warunkach licencji Creative Commons Uznanie autorstwa 3.0

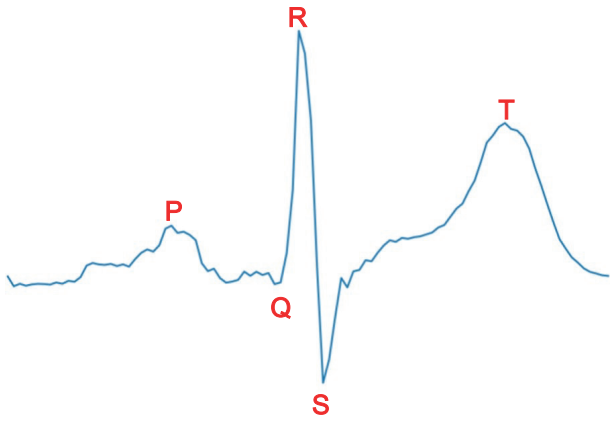

Fig. 1. Example of ECG waveform
Rys. 1. Przykładowy fragment przebiegu EKG

At the core of ECG signal analysis there are shapes from individual heartbeats, morphological features of the ECG curve components, and their temporal properties. This process requires knowledge of the location and morphology of its various components, such as P-QRS-T. These features correspond to the location, duration, amplitude, and shape of individual waves or deflections. An important step in considering issues corresponding to the analysis of electrocardiographic signals is the pre-processing and de-noising of the signal [16]. The next step is feature extraction, which involves extracting characteristic information from the electrocardiographic waveform. The next step is the ECG signal classification, corresponding to signal evaluation in the context of cardiac diseases. The realization of these areas is possible mainly due to the availability of large, public open-source ECG datasets. The most commonly used databases are Physionet, PhysioBank, and PhysioToolkit [3].

Remote monitoring systems in medicine are seen as a composite of three components: a device (which collects data on physiological parameters), a network-communication interface (which enables data transmission), and a remote analytical platform (which collects and integrates large amounts of data, often enriched with pre-processing and analysis algorithms) [17]. Measurements are most often realized through devices placed on/near the human body. The sensors used for this purpose, in a non-invasive and convenient way for the patient, can provide real-time continuous recording of relevant physiological parameters over a long period of time.

From a clinical and research point of view, ideally, ECG signal monitoring devices should combine the features of an ECG Holter device with the simultaneous ability to transmit and analyze data in real time. The new devices should allow the implementation of novel diagnostic algorithms to find patterns and correlate with standards accepted in the diagnosis and treatment of heart disease. This corresponds to classification and compression processes implemented within the device, as well as within the network and medical system. In this context, the use of artificial intelligence techniques works well. The literature indicates using different approaches to ECG signal classification using machine learning (ML) methods [4]. This division includes classical machine learning and deep learning methods. The most commonly mentioned involve the use of artificial neural networks [12, 13, 23], the Support Vector Machine [10], the Decision Tree [9], the K-Nearest-Neighbor [11], the Multi-Layer-Perceptron, the Deep Learning methods [1, 6, 14, 21, 22], hybrid methods [7, 18] and many others.

An important area from the perspective of the ECG signal processing and analysis is its application in everyday life solutions. This is possible due to the development and proliferation of devices for remote monitoring vital signs. Mobile solutions, i.e. small and unobtrusive devices, are increasingly used for long-term ECG monitoring and prevention of heart disease complications. Furthermore, increasingly associated with this context is the Internet of Things (IoT) paradigm, related to smart sensor networks and wireless communication protocols. From the perspective of the high incidence of heart disease, as well as the patients in underdeveloped countries, such solutions could provide automatic analysis of the ECG signal and, for example, classification of various heart diseases [5].

Considering the development of wireless and remote data transmission technologies, LoRa communication technology may find use. LoRa is a standard for long-distance, low-power communication within the IoT [19]. IoT defines a network of interconnected objects (objects), each with a unique identifier (address) and the ability to indirectly or directly store, process or exchange data [15]. A large number of IoT devices use technologies such as GSM, Wi-Fi, ZigBee or Bluetooth to transmit data [8]. However, none of them provide long range, with low power consumption and no transmission fees at the same time. Technologies such as Bluetooth, ZigBee, and Wi-Fi have severely limited range. GSM technology is a long-range and relatively high-bandwidth solution, but its use is associated with high energy requirements and the need to pay additional fees. LoRa technology, on the other hand, is characterized by very long range (more than 10 km) and low energy consumption at the expense of very low throughput. LoRa technology makes it possible to create your own network by erecting one or more LoRa gateways, allowing LoRa devices to communicate with, for example, a server. It is also possible to use The Things Network (TTN), which is a public network created and maintained by the community. This solution does not involve additional fees but has limitations on the amount of data transferred. Since it is a community-maintained network, it does not have as much coverage as, for example, commercial GSM networks, while it is possible to put up and plug into TTN your own LoRa gateway.

This paper presents a system for remote monitoring of ECG signal enriched with the results of studies of the application of machine learning models from the group of Tree-based ML techniques and Neural Networks in the context of heart disease classification. The idea of the system is a mobile recorder for monitoring the electrical activity of the heart muscle continuously for 24 hours or longer. The solution provides simultaneous analysis of rhythm and heart rate and controls in real time the phenomena occurring within the myocardium. The recorded data can be transmitted to the medical facility and/or directly to the attending physician. For the input data, preprocessing based on aggregation and feature extraction was performed using the Orthogonal Matching Pursuit (OMP) method. In addition, the performance of various models, combinations of input data (both raw and extracted data) was examined. Proposals for models and a set of features for input data relevant to classification were developed. Lastly, a verification of the applicability of the proposed methods on IoT devices has been made.

2. ECG LoRa devices

2.1. The system architecture

The ECG LoRa system includes devices, and a server in a medical facility or the cloud. Communication with the devices is carried out over a LoRa network (fig. 2). It can be either a private network, such as a single gateway connected to a server in a senior citizen's home, or a public network, such as The Things Network. A private network, on the one hand, has a limited range, covering no more than a few kilometers, but no transmission restrictions. A public network, on the other hand, has no coverage limitations (although coverage is not present everywhere), while it has limitations on the amount of data transmitted with a single device.

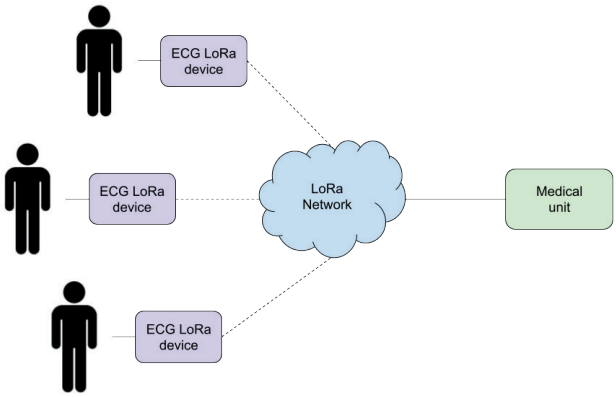

Fig. 2. Diagram of a system based on ECG LoRa device
Rys. 2. Schemat działania systemu opartego o urządzenie ECG LoRa

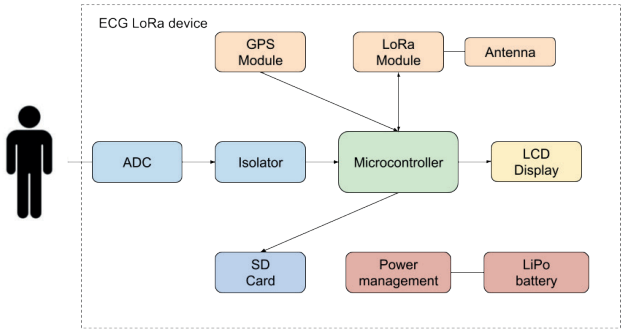

Fig. 3. Block diagram of ECG LoRa device
Rys. 3. Schemat blokowy urządzenia ECG LoRa

2.2. The device architecture

The ECG LoRa device includes (fig. 3):

- Microcontroller ESP32-S3 – 32-bit 2-core microcontroller equipped with Wi-Fi module, operating at 240 MHz clock; responsible for logic of system operation, handling of other devices, classification and compression of ECG signal.
- LoRa module with antenna – communication module responsible for transferring data to and from the medical facility via the LoRa network; realized on the SX1276 chip.
- ADC – 16-bit analog front-end for biopotential measurements; realized on the ADS1194CPAG chip, an analog-to-digital converter with SPI interface.
- Isolator – reinforced digital isolator with integrated power; isolates the patient from the rest of the circuit (isolates the SPI bus, and the ADC power supply).
- GPS module – a L80M39 module, turned on when needed; used for patient location and time synchronization.
- User interface consisting of LCD display, RGB LED and two buttons.
- SD Card – a card used to record the measured ECG signal, logs and device configuration.
- Power management and LiPo battery – a module that allows charging via USB and manages LiPo battery operation.
- Electrodes connectors – connectors for electrodes.

Figures 4 and 5 show the PCB and layout of the ECG LoRa device components.

3. Materials and Methods of ECG classification

The methodology used in the study was as follows (fig. 6). Records from the PTB-XL database were used for the study. The R-waves of the ECG waveforms were then detected and labeled. On this basis, QRS complexes were extracted. The data was divided into data for teaching Orthogonal Matching Pursuit (OMP) dictionaries and data used in the training/validation/test process of the models. Cross-validation was used, that is, the training/validation/test data was split using five different seeds. Three types of OMP dictionaries were created. Then, for the data intended for training/testing the models, OMP atom coefficients were obtained. The following data was combined to train/test the models.

The following data was combined to train/test the models:

— QRS complexes,
— OMP atom coefficients,
— Metadata.

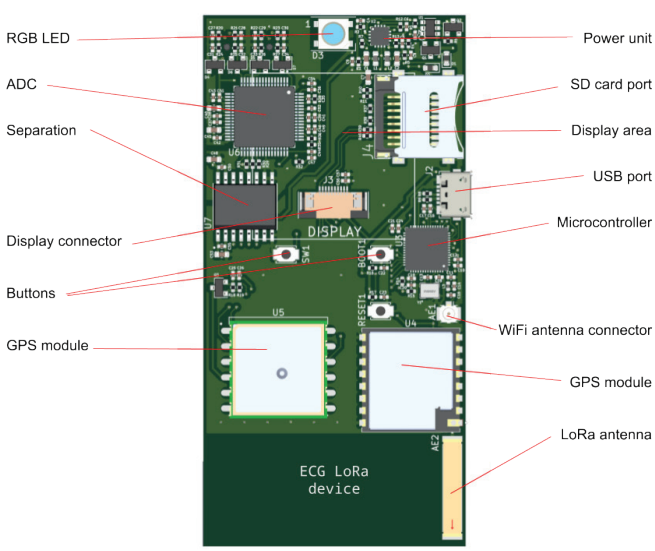

Fig. 4. ECG LoRa device elements layout, top side
Rys. 4. Rozmieszczenie elementów urządzenia ECG LoRa, strona górna

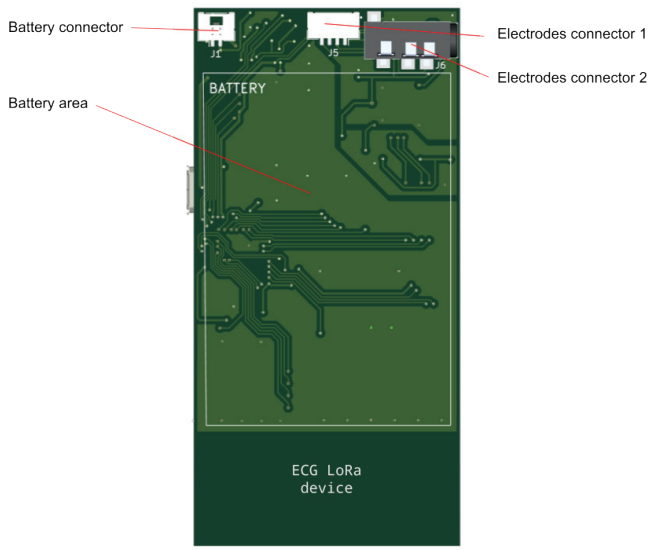

Fig. 5. ECG LoRa device elements layout, bottom side
Rys. 5. Rozmieszczenie elementów urządzenia ECG LoRa, strona dolna

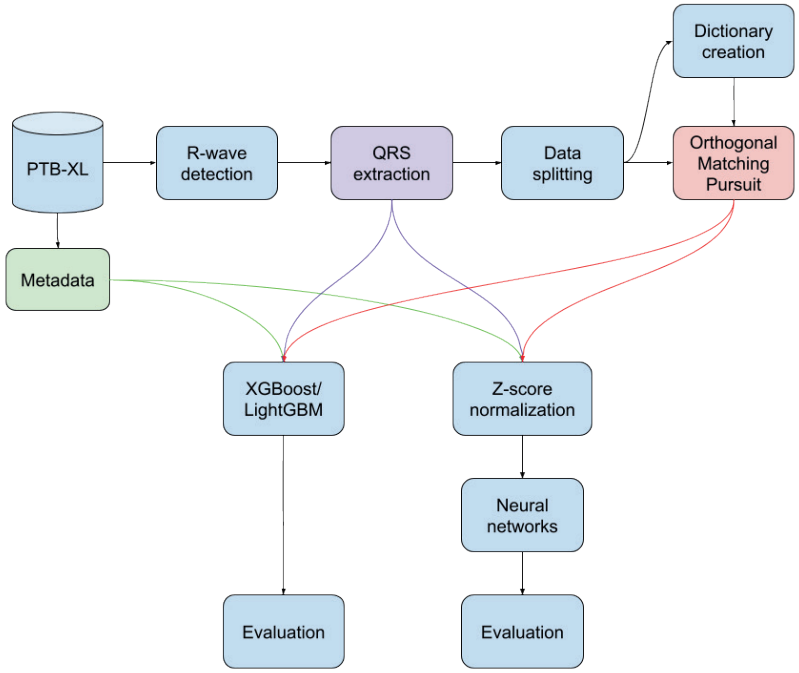

Fig. 6. A general overview diagram of the method
Rys. 6. Ogólny schemat metody

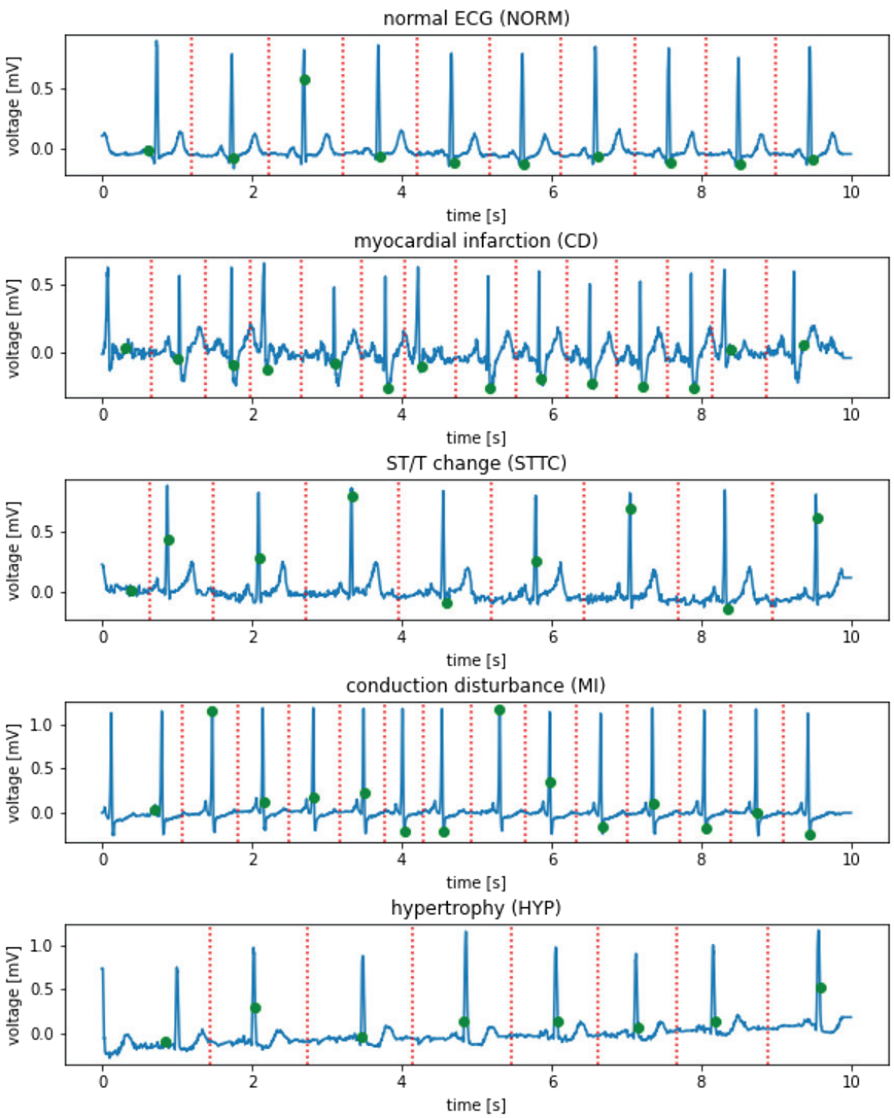

Fig. 7. An example of the ECG waveform for each class, along with the detected R-waves (green) and cut lines (red)
Rys. 7. Przykładowy przebieg EKG dla każdej z klas wraz z wykrytymi załamkami R (zielony) i liniami cięć (czerwony)

Then two different approaches were used:
– Tree-based ML techniques,
– Neural Networks only.

The study consisted of classifying 2, 5, and 15 cardiovascular diseases. In the last step, an evaluation was made.

3.1. PTB-XL Dataset

The research for the study was carried out using the PTB-XL database [3, 24], a collection of electrocardiographic signals recorded for 12 leads. The dataset identified 21,837 ECG records, 10 s in length, collected from 18,885 individuals (men and women). The dataset was balanced between both sexes. However, it was not balanced regarding the number of included records in classes and subclasses for the study. Records that lacked labels (in the sense of the assigned class) and records for which the certainty of medical diagnosis assignment was not 100 % were excluded. This resulted in 17,011 records that referred to one of two classes (NORM class and others from the database), five classes (disease classes from the database), and 15 subclasses (disease subclasses). A detailed description of the classes and subclasses is included in the source literature reference of the database creators. ECG recordings with a sampling rate of 500 Hz with 16-bit resolution were used for the study.

3.2. QRS extraction

The method described in [20] was used to extract QRS complexes. This method labels R-waves in the first step. For this purpose, it uses known R-wave detectors (Two Average, Christov, Engzee) on all 12 leads and then, using the k-mean algorithm, deduces the true position of R-waves. In the next stage, the centers of the segments between the R-waves are determined which represent the location of the signal cut. With this approach, the R-waves are at or very close to the center of the extracted QRS complexes. In a further step, the determined R-peaks were used to cut the 10-second recordings into ECG segments for each of the five classes of heart disease separately (fig. 7).

3.3. Data splitting

At this stage, the experience of the article [20] was used. However, the parameter combinations used in [20] were limited:
– Dictionaries (for OMP algorithm): Dictionary Learning (DL), KSVD, Gabor dictionary,
– Dictionary sizes: 125, 250 elements,
– Number of non-zero coefficients: 20,
– Aggregation methods: Voting.

In this way, collections of records were obtained, each record contained the following types of data:
– Metadata (sex, age, BPM, resampling ratio),
– Segments containing QRS complexes,
– Coefficients from the OMP algorithm,
– Classes and subclasses of heart disease from the PTB-XL database.

Each set of records was divided into training (70 %), validation (15 %), and test (15 %) data. To get a more accurate assessment of the model's performance, non-exhaustive cross-validation was used. The split function was used with five different seeds to repeat all tests five times for different data splits. For statistical significance tests, the results were recalculated for the test data using a split function with 50 different seeds.

3.4. Orthogonal Matching Pursuit

The issue of classification using machine learning methods have been pursued by other researchers in various combinations. The presented work proposes a method for classifying cardiac arrhythmias based on a multi-lead ECG signal. There are methods based on neural networks, however, to the knowledge of the authors of this work, they can be enhanced by the Orthogonal Matching Pursuit (OMP) method. The OMP is a technique that can provide an optimal representation of the ECG signal, defined as a subset of dictionary elements, as is the case with the ECG signal representations as linear expansions relative to a specified set of basic functions, well localized in time and/or frequency. For dictionaries, the OMP linear combination explains the largest percentage of the signal energy among all subsets of the same amount. For the OMP, dictionaries created using Dictionary Learning (DL) Technique, KSVD Technique, and Gabor functions were created. The DL and KSVD dictionaries were based on data, i.e., the ECG signal waveforms contained in the PTB-XL database. A detailed description of the Orthogonal Matching Pursuit method, as well as the dictionary creation, is described in the paper [20].

3.5. Tree-based ML techniques

In this approach, the optimized XGBoost, and LightGBM classifiers were used for classification (fig. 8). The following parameters were changed:
– Boosting learning rate (learning_rate) – 0.01, 0.1, 0.3,
– Number of boosting rounds (n_estimators) – 100, 200,
– Subsample ratio of columns when constructing each tree (colsample_bytree) – 0.6, 1.0,
– Subsample ratio of the training instance (subsample) – 0.6, 1.0.

QRS complexes, OMP atomic coefficients (obtained from three types (Gabor, DL, KSVD), and two sizes (125, 250 atoms) of dictionaries), and metadata were provided at the model inputs, depending on the study. In this case, all data types were flattened to a 1-dimensional array. Since the ECG records studied had different numbers of R-waves, and thus different numbers of QRS complexes while the model had a fixed number of inputs, an aggregation method called Voting was used [20]. It is based on the fact that for each QRS complex the classification is done independently, and then the probabilities from all of them are averaged, and the final result is determined on this basis.

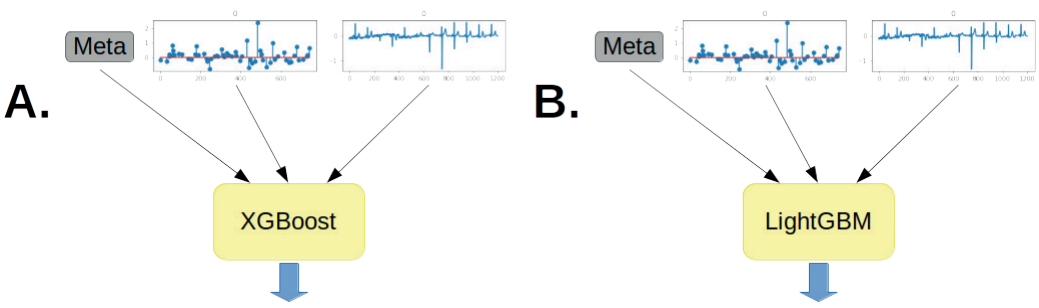

Fig. 8. Tree-based ML techniques; A. XGBoost, B. LightGBM
Rys. 8. Techniki uczenia maszynowego oparte na drzewach; A. XGBoost, B. LightGBM

3.6. Neural Networks

This approach uses six artificial neural network architectures, referred to by the authors as VanillaNet (VN), BigVanillaNet (BVN), ConvolutionNet1 (CN1), ConvolutionNet2 (CN2), ConvolutionNet3 (CN3), ConvolutionNet4 (CN4). VanillaNet and BigVanillaNet (fig. 9) are networks consisting of linear layers. Their parameters are presented in tables 1 and 2. ConvolutionNet (fig. 10) additionally contains convolutional layers. Their parameters are presented in tables 3 to 8.

Table 1. VanilaNet network architecture. n_inputs – input tensor size, n_outputs – output tensor size
Tabela 1. Architektura sieci VanilaNet. n_inputs – wielkość tensora wejściowego, n_outputs – wielkość tensora wyjściowego

Layer	Inputs	Outputs
Linear 1	n_inputs	50
Linear 2	50	20
Linear 3	20	n_outputs

Table 2. BigVanilaNet network architecture. n_inputs – input tensor size, n_outputs – output tensor size
Tabela 2. Architektura sieci BigVanilaNet. n_inputs – wielkość tensora wejściowego, n_outputs – wielkość tensora wyjściowego

Layer	Inputs	Outputs
Linear 1	n_inputs	200
Linear 2	200	100
Linear 3	100	50
Linear 4	50	20
Linear 5	20	n_outputs

As with tree-based models, the combinations of QRS complexes, OMP atomic coefficients, and metadata were given as

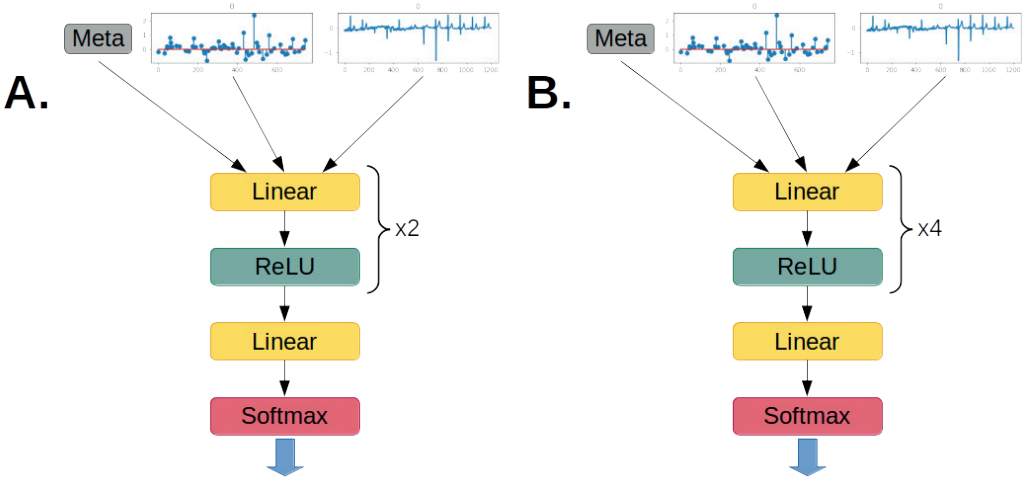

Fig. 9. Neural Network architectures; A. VanillaNet, B. BigVanillaNet
Rys. 9. Architektury sieci neuronowych; A. VanillaNet, B. BigVanillaNet

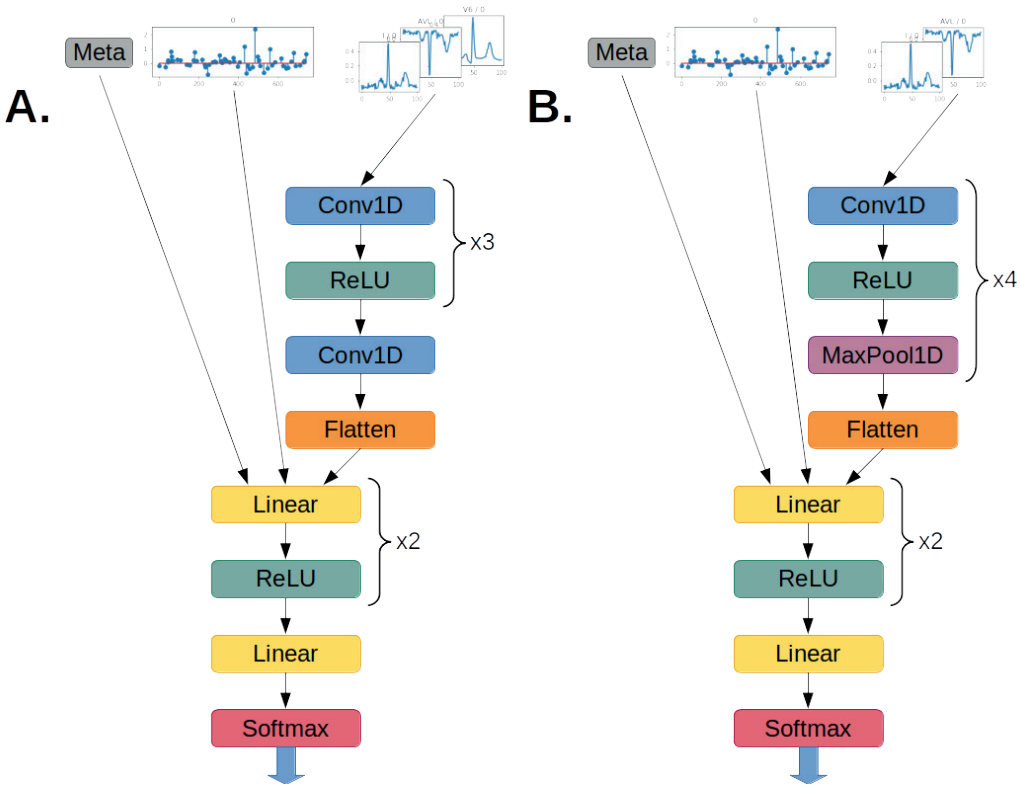

Fig. 10. Neural Network architectures; A. ConvolutionNet1 and ConvolutionNet2, B. ConvolutionNet3 or ConvolutionNet4
Rys. 10. Architektury sieci neuronowych; A. ConvolutionNet1 and ConvolutionNet2, B. ConvolutionNet3 or ConvolutionNet4

Table 3. ConvolutionNet1 network architecture – convolutional layers
Tabela 3. Architektura sieci ConvolutionNet1 – warstwy konwolucyjne

Layer	Channels in	Channels out	Kernel size	Stride	Padding	Inputs	Outputs
Conv1D 1	12	24	3	2	1	12 × 100	24 × 50
Conv1D 2	24	48	3	2	1	24 × 50	48 × 25
Conv1D 3	48	96	3	2	1	48 × 25	96 × 13
Conv1D 4	96	96	3	1	1	96 × 13	96 × 13

Table 4. ConvolutionNet2 network architecture – convolutional layers
Tabela 4. Architektura sieci ConvolutionNet2 – warstwy konwolucyjne

Layer	Channels in	Channels out	Kernel size	Stride	Padding	Inputs	Outputs
Conv1D 1	12	24	5	2	2	12 × 100	24 × 50
Conv1D 2	24	48	5	2	2	24 × 50	48 × 25
Conv1D 3	48	96	5	2	2	48 × 25	96 × 13
Conv1D 4	96	96	5	1	2	96 × 13	96 × 13

Table 5. ConvolutionNet3 network architecture – convolutional layers
Tabela 5. Architektura sieci ConvolutionNet3 – warstwy konwolucyjne

Layer	Channels in	Channels out	Kernel size	Stride	Padding	Inputs	Outputs
Conv1D 1	12	12	3	1	1	12 × 100	12 × 100
MaxPool1D 1	–	–	2	0	1	12 × 100	12 × 50
Conv1D 2	12	12	3	1	1	12 × 50	12 × 50
MaxPool1D 2	–	–	2	0	1	12 × 50	12 × 25
Conv1D 3	12	12	3	1	1	12 × 25	12 × 25
MaxPool1D 3	–	–	2	0	1	12 × 25	12 × 12
Conv1D 4	12	12	3	1	1	12 × 12	12 × 12
MaxPool1D 4	–	–	2	0	1	12 × 12	12 × 6

Table 6. ConvolutionNet4 network architecture – convolutional layers
Tabela 6. Architektura sieci ConvolutionNet4 – warstwy konwolucyjne

Layer	Channels in	Channels out	Kernel size	Stride	Padding	Inputs	Outputs
Conv1D 1	12	12	5	1	2	12 × 100	12 × 100
MaxPool1D 1	–	–	2	0	1	12 × 100	12 × 50
Conv1D 2	12	12	5	1	2	12 × 50	12 × 50
MaxPool1D 2	–	–	2	0	1	12 × 50	12 × 25
Conv1D 3	12	12	5	1	2	12 × 25	12 × 25
MaxPool1D 3	–	–	2	0	1	12 × 25	12 × 12
Conv1D 4	12	12	5	1	2	12 × 12	12 × 12
MaxPool1D 4	–	–	2	0	1	12 × 12	12 × 6

Table 7. ConvolutionNet1 and ConvolutionNet2 network architecture – linear layers. n_inputs – input tensor size, n_outputs – output tensor size
Tabela 7. Architektura sieci ConvolutionNet1 i ConvolutionNet2 – warstwy liniowe. n_inputs – wielkość tensora wejściowego, n_outputs – wielkość tensora wyjściowego

Layer	Inputs	Outputs
Linear 1	n_inputs + 96 × 13	50
Linear 2	50	20
Linear 3	20	n_outputs

Table 8. ConvolutionNet3 and ConvolutionNet4 network architecture – linear layers, n_inputs – input tensor size, n_outputs – output tensor size
Tabela 8. Architektura sieci ConvolutionNet3 i ConvolutionNet4 – warstwy liniowe, n_inputs – wielkość tensora wejściowego, n_outputs – wielkość tensora wyjściowego

Layer	Inputs	Outputs
Linear 1	n_inputs + 96 × 13	50
Linear 2	50	20
Linear 3	20	n_outputs

input. However, for VanillaNet and BigVanillaNet networks, all data types were flattened to a 1-dimensional tensor. For ConvolutionNet networks, the OMP atom coefficients and metadata were flattened and the QRS complexes were given without changes on the convolution layers. The neural networks were trained using cross entropy as a loss function, the Adam optimizer with lr = 0.001, and took 100 epochs. As in previous approaches, the voting aggregation method, as well as the dictionaries were used.

3.7. Metrics

The models were evaluated using the following metrics [20]:
- Accuracy. It is calculated as follows: Accuracy = (True Positives + True Negatives) / (Total Predictions).
- Precision. It is calculated as follows: Precision = True Positives / (True Positives + False Positives).
- Recall. It is calculated as follows: Recall = True Positives / (True Positives + False Negatives).
- F1 score. It is calculated as follows: F1 = 2 · Precision · Recall / (Precision + Recall).
- Balanced accuracy: Balanced Accuracy = (True Positives / (True Positives + False Negatives)) + (True Negatives / (True Negatives + False Positives)).

3.8. Used tools

Two types of servers were used to perform the calculations:
- one GPU server with 2 Intel Xeon Silver 4210R processors, 192 GB of RAM, and NVIDIA Tesla A100 and NVIDIA Tesla A40 cards,
- five CPU servers with 2 Intel Xeon Gold 6132 processors and 512 GB of RAM each.

The following tools were used as software: JupyterLab, PyTorch, Sklearn, Numpy and Pandas.

4. Results

All experiments were conducted for 2, 5, and 15 disease classes. The following dictionaries were used for the research: Gabor, Dictionary Learning, and KSVD. 125 and 250 element dictionaries were used. Various combinations of input data were used: QRS complexes, OMP atomic coefficients, metadata, further labeled: signal, coef, meta. All tests were repeated for five seeds of data splitting into training, validation, and test data. The above parameters are identical in all approaches.

Table 9. First decile of tree-based classification results calculated in terms of ACC for validation data – two classes
Tabela 9. Pierwszy decyl wyników klasyfikacji opartej na drzewie obliczony pod względem ACC dla danych walidacyjnych – dwie klasy

Model	Dictionary	lr	ne	cb	sb	Acc	Precision	Recall	F1	BAcc
LGBM	gabor-125	0.3	200	1.0	1.0	0.908	0.904	0.908	0.906	0.908
LGBM	gabor-125	0.3	200	1.0	0.6	0.908	0.904	0.908	0.906	0.908
LGBM	gabor-250	0.3	200	1.0	0.6	0.907	0.903	0.907	0.905	0.907
LGBM	gabor-250	0.3	200	1.0	1.0	0.907	0.903	0.907	0.905	0.907
XGBoost	gabor-250	0.3	200	1.0	1.0	0.907	0.903	0.907	0.905	0.907
XGBoost	gabor-125	0.3	200	0.6	1.0	0.907	0.903	0.907	0.905	0.907
XGBoost	gabor-125	0.3	200	1.0	1.0	0.907	0.903	0.907	0.904	0.907
XGBoost	gabor-250	0.3	200	0.6	0.6	0.907	0.903	0.906	0.904	0.906
XGBoost	gabor-125	0.3	100	1.0	1.0	0.906	0.903	0.906	0.904	0.906
LGBM	gabor-125	0.3	200	0.6	1.0	0.906	0.903	0.906	0.904	0.906
LGBM	gabor-125	0.3	200	0.6	0.6	0.906	0.903	0.906	0.904	0.906
XGBoost	gabor-125	0.3	200	1.0	0.6	0.906	0.903	0.906	0.904	0.906
LGBM	gabor-250	0.3	200	0.6	0.6	0.906	0.903	0.906	0.904	0.906
LGBM	gabor-250	0.3	200	0.6	1.0	0.906	0.903	0.906	0.904	0.906

4.1. Results for Tree-based ML techniques

In this approach, due to a large number of combinations of optimization parameters, only signal+coef+meta was used as the input data type. Tables 9, 10 and 11 contain the first decile of classification results calculated in terms of ACC for validation data.

Table 10. First decile of ML based classification results calculated in terms of ACC for validation data – five classes
Tabela 10. Pierwszy decyl wyników klasyfikacji opartej na drzewie obliczony pod względem ACC dla danych walidacyjnych – pięć klas

Model	Dictionary	lr	ne	cb	sb	Acc	Precision	Recall	F1	BAcc
LGBM	gabor-125	0.3	200	1.0	0.6	0.783	0.746	0.693	0.713	0.693
LGBM	gabor-125	0.3	200	1.0	1.0	0.783	0.746	0.693	0.713	0.693
LGBM	gabor-250	0.3	200	1.0	0.6	0.782	0.749	0.693	0.714	0.693
LGBM	gabor-250	0.3	200	1.0	1.0	0.782	0.749	0.693	0.714	0.693
LGBM	gabor-250	0.3	200	0.6	0.6	0.782	0.748	0.693	0.714	0.693
LGBM	gabor-250	0.3	200	0.6	1.0	0.782	0.748	0.693	0.714	0.693
LGBM	gabor-250	0.1	200	1.0	1.0	0.782	0.753	0.696	0.718	0.696
LGBM	gabor-250	0.1	200	1.0	0.6	0.782	0.753	0.696	0.718	0.696
LGBM	gabor-125	0.3	100	1.0	1.0	0.781	0.743	0.693	0.712	0.693
LGBM	gabor-125	0.3	100	1.0	0.6	0.781	0.743	0.693	0.712	0.693
LGBM	gabor-125	0.3	200	0.6	1.0	0.780	0.745	0.692	0.712	0.692
XGBoost	gabor-125	0.3	200	0.6	0.6	0.780	0.745	0.692	0.712	0.692
LGBM	gabor-250	0.3	100	0.6	0.6	0.780	0.747	0.694	0.715	0.694
LGBM	gabor-250	0.3	100	0.6	1.0	0.780	0.747	0.694	0.715	0.694

Table 11. First decile of ML based classification results calculated in terms of ACC for validation data – 15 classes
Tabela 11. Pierwszy decyl wyników klasyfikacji opartej na drzewie obliczony pod względem ACC dla danych walidacyjnych – 15 klas

Model	Dictionary	lr	ne	cb	sb	Acc	Precision	Recall	F1	BAcc
LGBM	gabor-250	0.1	200	1.0	0.6	0.714	0.579	0.451	0.457	0.451
LGBM	gabor-250	0.1	200	1.0	1.0	0.714	0.579	0.451	0.457	0.451
LGBM	gabor-250	0.1	200	0.6	0.6	0.714	0.571	0.449	0.457	0.449
LGBM	gabor-250	0.1	200	0.6	1.0	0.714	0.571	0.449	0.457	0.449
LGBM	gabor-250	0.1	100	0.6	0.6	0.712	0.559	0.453	0.461	0.453
LGBM	gabor-250	0.1	100	0.6	1.0	0.712	0.559	0.453	0.461	0.453
XGBoost	gabor-250	0.3	200	1.0	1.0	0.712	0.571	0.447	0.453	0.447
LGBM	gabor-250	0.1	100	1.0	1.0	0.712	0.564	0.451	0.457	0.451
LGBM	gabor-250	0.1	100	1.0	0.6	0.712	0.564	0.451	0.457	0.451
XGBoost	gabor-250	0.3	200	0.6	1.0	0.712	0.608	0.447	0.455	0.447
LGBM	gabor-125	0.1	200	0.6	1.0	0.711	0.562	0.446	0.450	0.446
LGBM	gabor-125	0.1	200	0.6	0.6	0.711	0.562	0.446	0.450	0.446
XGBoost	gabor-250	0.3	200	1.0	0.6	0.711	0.580	0.445	0.453	0.445
XGBoost	gabor-250	0.3	200	0.6	0.6	0.711	0.586	0.444	0.449	0.444

4.1. Results for Neural Networks

Tables 12, 13 and 14 contain the best results for each combination of data input and neural network models in terms of ACC for validation data. As it can be seen, the ConvolutionalNet models have the highest accuracy for classifying 2, 5 and 15 classes. The addition of the MaxPool1D layer in the convolu-

Table 12. Best results for each combination of data input and neural network models in terms of ACC for validation data – two classes
Tabela 12. Najlepsze wyniki dla każdej kombinacji danych wejściowych i modeli sieci neuronowych pod względem ACC dla danych walidacyjnych – dwie klasy

Model	Data type	Dictionary	Acc	Precision	Recall	F1	BAcc
CN3	signal	N/A	0.905	0.901	0.905	0.903	0.905
CN1	signal-coef-meta	gabor-125	0.905	0.901	0.904	0.902	0.904
CN2	signal-coef	gabor-125	0.905	0.901	0.904	0.902	0.904
CN2	signal-coef-meta	gabor-125	0.904	0.901	0.903	0.902	0.903
CN3	signal-meta	N/A	0.904	0.900	0.903	0.901	0.903
CN4	signal-meta	N/A	0.903	0.900	0.901	0.901	0.901
CN4	signal-coef	gabor-125	0.903	0.899	0.902	0.901	0.902
CN4	signal-coef-meta	gabor-125	0.902	0.898	0.902	0.900	0.902
CN4	signal	N/A	0.902	0.898	0.901	0.899	0.901
CN1	signal-coef	gabor-125	0.901	0.898	0.901	0.899	0.901
BVN	signal-coef	gabor-125	0.901	0.897	0.900	0.898	0.900
BVN	signal-coef-meta	gabor-125	0.901	0.897	0.901	0.899	0.901
CN3	signal-coef-meta	ksvd-125	0.901	0.897	0.901	0.899	0.901
VN	signal-coef-meta	ksvd-125	0.900	0.896	0.899	0.897	0.899
CN3	signal-coef	gabor-125	0.899	0.896	0.899	0.897	0.899
BVN	signal-meta	N/A	0.899	0.895	0.900	0.897	0.900
VN	signal-meta	N/A	0.899	0.896	0.898	0.897	0.898
BVN	signal	N/A	0.899	0.895	0.898	0.896	0.898
VN	signal	N/A	0.898	0.894	0.899	0.896	0.899
BVN	coef-meta	ksvd-125	0.898	0.894	0.897	0.895	0.897
VN	coef-meta	ksvd-125	0.897	0.893	0.897	0.895	0.897
VN	signal-coef	ksvd-125	0.897	0.893	0.898	0.895	0.898
VN	coef	ksvd-250	0.895	0.892	0.894	0.893	0.894
BVN	coef	ksvd-250	0.895	0.891	0.895	0.893	0.895
CN	signal-meta	N/A	0.847	0.859	0.837	0.838	0.837
CN2	signal-meta	N/A	0.840	0.842	0.836	0.833	0.836
CN2	signal	N/A	0.783	0.723	0.766	0.737	0.766
CN1	signal	N/A	0.757	0.822	0.733	0.695	0.733
VN	meta	N/A	0.727	0.721	0.709	0.712	0.709
BVN	meta	N/A	0.722	0.716	0.704	0.707	0.704

tion layers has improved the accuracy (ConvolutionalNet3 and ConvolutionalNet4). Both kernel_size = 3 and kernel_size = 5 in Conv1D layers bring good results. The best result is obtained through the use of Gabor and KSVD dictionaries. DL dictionary does not provide a similar result.

Table 13. Best results for each combination of data input and neural network models in terms of ACC for validation data – five classes
Tabela 13. Najlepsze wyniki dla każdej kombinacji danych wejściowych i modeli sieci neuronowych pod względem ACC dla danych walidacyjnych – pięć klas

Model	Data type	Dictionary	Acc	Precision	Recall	F1	BAcc
CN4	signal-coef-meta	gabor-125	0.766	0.597	0.605	0.597	0.605
CN4	signal-coef	gabor-125	0.766	0.671	0.648	0.655	0.648
CN1	signal-coef-meta	gabor-250	0.766	0.593	0.604	0.596	0.604
CN3	signal-coef-meta	gabor-125	0.764	0.588	0.604	0.594	0.604
CN2	signal-coef-meta	ksvd-125	0.763	0.591	0.602	0.594	0.602
CN1	signal-coef	ksvd-125	0.762	0.591	0.598	0.592	0.598
CN3	signal-coef	dl-125	0.761	0.619	0.602	0.597	0.602
CN3	signal-meta	N/A	0.760	0.587	0.599	0.591	0.599
CN2	signal-coef	ksvd-125	0.758	0.587	0.597	0.588	0.597
CN4	signal-meta	N/A	0.755	0.591	0.591	0.586	0.591
CN3	signal	N/A	0.753	0.582	0.591	0.582	0.591
VN	signal-coef	ksvd-125	0.746	0.634	0.603	0.607	0.603
VN	signal-coef+meta	gabor-125	0.746	0.574	0.579	0.572	0.579
VN	coef-meta	ksvd-125	0.745	0.640	0.593	0.599	0.593
VN	signal-meta	N/A	0.741	0.569	0.577	0.570	0.577
VN	coef	ksvd-125	0.737	0.587	0.570	0.568	0.570
VN	signal	N/A	0.736	0.567	0.572	0.565	0.572
BVN	coef	ksvd-125	0.734	0.654	0.571	0.573	0.571
BVN	coef-meta	ksvd-125	0.731	0.563	0.561	0.557	0.561
BVN	signal	N/A	0.726	0.608	0.569	0.565	0.569
CN4	signal	N/A	0.724	0.564	0.566	0.556	0.566
BVN	signal-coef-meta	ksvd-125	0.721	0.630	0.568	0.575	0.568
BVN	signal-coef	ksvd-250	0.721	0.551	0.555	0.549	0.555
BVN	signal-meta	N/A	0.719	0.584	0.555	0.555	0.555
CN1	signal	N/A	0.575	0.394	0.392	0.358	0.392
CN1	signal-meta	N/A	0.517	0.376	0.323	0.281	0.323
CN2	signal-meta	N/A	0.499	0.267	0.315	0.259	0.315
VN	meta	N/A	0.466	0.296	0.292	0.279	0.292
BVN	meta	N/A	0.457	0.285	0.287	0.274	0.287
CN2	signal	N/A	0.334	0.125	0.221	0.119	0.221

Table 14. Best results for each combination of data input and neural network models in terms of ACC for validation data – 15 classes
Tabela 14. Najlepsze wyniki dla każdej kombinacji danych wejściowych i modeli sieci neuronowych pod względem ACC dla danych walidacyjnych – 15 klas

Model	Data type	Dictionary	Acc	Precision	Recall	F1	BAcc
CN3	signal-coef	ksvd-125	0.680	0.346	0.376	0.356	0.376
CN4	signal-coef	gabor-250	0.677	0.359	0.388	0.366	0.388
CN4	signal-coef-meta	dl-125	0.676	0.339	0.377	0.353	0.377
CN3	signal-coef-meta	gabor-125	0.672	0.342	0.382	0.355	0.382
VN	signal-coef-meta	ksvd-250	0.659	0.340	0.367	0.347	0.367
VN	signal-coef	gabor-125	0.656	0.351	0.357	0.341	0.357
VN	coef-meta	ksvd-250	0.654	0.348	0.358	0.344	0.358
VN	coef	ksvd-250	0.652	0.364	0.356	0.346	0.356
VN	signal	N/A	0.650	0.306	0.343	0.317	0.343
VN	signal-meta	N/A	0.648	0.298	0.345	0.315	0.345
CN3	signal-meta	N/A	0.634	0.231	0.287	0.252	0.287
CN4	signal-meta	N/A	0.624	0.229	0.290	0.250	0.290
BVN	coef-meta	ksvd-250	0.612	0.275	0.290	0.268	0.290
BVN	coef	ksvd-250	0.605	0.259	0.282	0.257	0.282
CN3	signal	N/A	0.601	0.225	0.263	0.228	0.263
BVN	signal-coef-meta	dl-125	0.590	0.212	0.259	0.230	0.259
CN4	signal	N/A	0.586	0.210	0.230	0.204	0.230
BVN	signal-coef	dl-250	0.584	0.217	0.240	0.220	0.240
BVN	signal	N/A	0.580	0.196	0.224	0.200	0.224
CN1	signal-coef-meta	ksvd-250	0.568	0.222	0.251	0.227	0.251
BVN	signal-meta	N/A	0.566	0.196	0.225	0.201	0.225
CN1	signal-coef	ksvd-250	0.467	0.104	0.132	0.110	0.132
VN	meta	N/A	0.431	0.078	0.083	0.064	0.083
BVN	meta	N/A	0.428	0.062	0.082	0.063	0.082
CN	signal	N/A	0.424	0.032	0.071	0.044	0.071
CN2	signal-coef-meta	dl-250	0.417	0.028	0.067	0.039	0.067

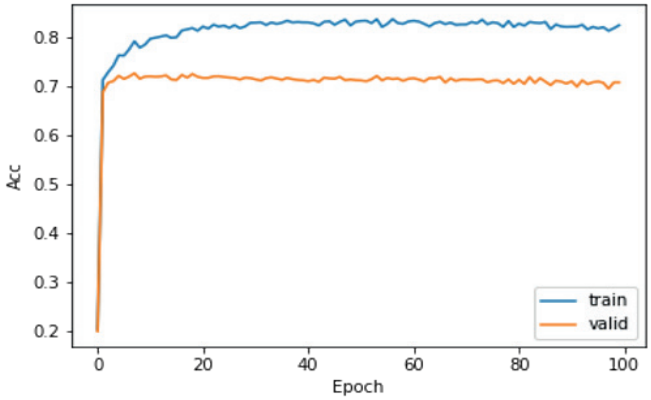

Fig. 11. Sample learning curve – Accuracy as a function of learning epoch
Rys. 11. Przykładowa krzywa uczenia – dokładność w funkcji epoki uczenia

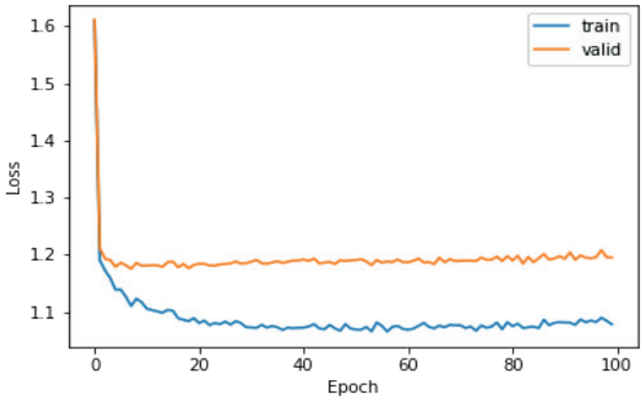

Fig. 12. Sample learning curve – loss as a function of learning epoch
Rys. 12. Przykładowa krzywa uczenia – strata w funkcji epoki uczenia

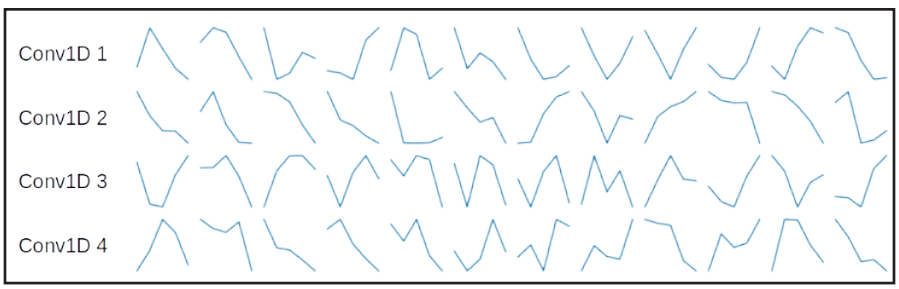

Fig. 13. Example filters of convolutional layers of the CN4 network
Rys. 13. Przykładowe filtry warstw konwolucyjnych sieci CN4

Figures 11 and 12 show an example of a learning curve (in this case for ConvolutionNet for five classes).

Figure 13 presents an example of filters for the convolutional layers of the CN4 network (which have kernel_size = 5).

4.3. Results for test data

Table 15 shows the results for the two presented approaches for the test data. The designations used in the tables correspond to the following descriptions: Classes – number of classes used in classification, Model – the classifier model that obtained the highest ACC score for a given approach, and the following columns are the evaluation metrics.

Evaluating Tree-based ML techniques based on the optimization of XGBoost and LightGBM classifiers, a slight improvement in the classification results for 2, 5, and 15 classes was noticed, compared to references [20]. In the paper [20], its authors used well-known classifier models with default parameters. An analysis of ACC metrics for the best items showed higher accuracy of 0.3 % for two classes, 0.6 % for five classes, and 0.4 % for 15 classes, respectively. Regardless of the number of classes considered, the LGBM classifier achieves better results than XGBoost. Evaluation of the optimization parameters suggests the validity of their verification and their common use in the undertaken studies. Keeping the default parameters of the LGBM or XGBoost models would render achieving this improvement impossible. It is worth noting that in every case Tree-based ML technique obtained better results for the test data than neural networks.

5. Verification of the applicability in the IoT

Verification of the applicability of the proposed models in the IoT devices has been carried out. As hardware for the research, ESP32-S3 microprocessor was used (dual-core Xtensa LX7 at 240 MHz, with 512 kB of Flash memory and 8 MB of RAM). As software, the micropython environment (which is a port of the python language to the microprocessor) and the ulab library (which is a port of the NumPy and SciPy libraries for Micropython) were used.

Table 15. Results for test data for Tree-Based ML techniques and Neural Networks
Tabela 15. Wyniki klasyfikacji technik ML opartych na drzewach i sieci neuronowych dla danych testowych

Classes	Model	Acc	Precision	Recall	F1	fused
2	LGBM	0.905	0.902	0.905	0.903	0.905
2	CN3	0.903	0.900	0.903	0.901	0.903
5	LGBM	0.783	0.743	0.692	0.711	0.692
5	CN4	0.763	0.592	0.604	0.594	0.604
15	LGBM	0.712	0.544	0.445	0.453	0.445
15	CN3	0.679	0.345	0.372	0.353	0.372

Table 16 shows the execution time of R-wave detectors for a 10-second 1-lead ECG signal sampled at 100 Hz. Table 17 shows the execution of features: OMP algorithm for 250-element dictionary, and convolution layers for CN3. Table 18 shows the execution time of Tree-based ML techniques. Table 19 shows the execution time of linear layers of chosen neural networks. Note that the trained LGBM model was too large to run on IoT devices.

The above ensures that the proposed methods on a 3-lead ECG signal, for selected models, will be able to perform the classification process in real time. Analysis of the 12-lead ECG signal requires skipping some of the QRS complex.

Table 16. Execution time of R-wave detectors for 10-second 1-lead ECG signal sampled at 100 Hz
Tabela 16. Czas wykonania detektorów załamka R dla 10-sekundowego 1-odprowadzeniowego sygnału EKG próbkowanego z częstotliwością 100 Hz

Detector	Time
Christov detector	849.809 ms
Engzee detector	604.709 ms
Two Average detector	844.207 ms

Table 17. Execution time of data processing for 1 lead ECG signal
Tabela 17. Czas wykonania przetwarzania danych dla 1 odprowadzenia

Feature	Time
OMP for 250-elements dictionary	227.787 ms
CN3 convolution layers	3705.104 ms

Table 18. Execution time of Tree-based ML techniques
Tabela 18. Czas wykonania technik ML opartych na drzewach

Feature	Data type	Time
XGBoost	signal+coef+meta	13.055 ms
LGBM	N/A	N/A

Table 19. Execution time of Neural Network
Tabela 19. Czas wykonania architektur sieci neuronowych

Feature	Data type	Time
VanilaNet	signal+coef+meta	247.386 ms
VanilaNet	signal	66.52 ms
VanilaNet	coef	170.075 ms
VanilaNet	meta	2.230 ms
VanilaNet	signal+coef	246.398 ms
VanilaNet	signal+meta	66.738 ms
VanilaNet	coef+meta	170.303 ms
CN3 linear layers	signal	4.722 ms
CN3 linear layers	signal+coef	172.302 ms
CN3 linear layers	signa+meta	4.918 ms
CN3 linear layers	signal+coef+meta	172.787 ms

6. Conclusion

The need for long-term ECG signal monitoring is increasingly recognized, despite the existence of advanced medical equipment in healthcare centers. This process carried out in real time, at home, should concern patients and the elderly with comorbidities. This is made possible by the advent of widely used portable ECG telemonitoring systems for recording arrhythmias as they occur or for recording ECGs for later observation of various trends.

The difficulties in applying cardiac telemonitoring can be boiled down to two issues. The first issue concerns real-time data analysis. This is a direct result of the increasing speed of data generation and the numerous devices that transmit data using sensors. This data should be analyzed in real time in order to take action on patient care. On the other hand, these aspects have to do with the processes involved in digital signal processing and compression, which is the second issue. Their limitations are related to the transmission bandwidth, memory and battery of the remote cardiac monitoring device and ECG recording. In such scenarios, the main goal right next to ECG signal compression is the issue of classification, which aims to evaluate the correlations present and interpret them in terms of determining the occurrence of heart disease, in a way that allows immediate action to be taken to reduce the risk of patient death. In addition, medical data collected from different patients over time is an important dataset that can be used to train machine learning models for smarter and targeted disease prediction and treatment.

The aforementioned difficulties can be overcome by properly designing the device, as well as enriching the device with appropriate methods for processing and analyzing the recorded signals. The use of LoRa technology in the device allows to reduce the cost of operating of the system significantly due to the avoidance of transmission fees. Moreover, unlike technologies such as GSM or even Wi-Fi, LoRa technology has very low power consumption. A limitation of LoRa technology is the impossibility of transmitting the ECG signal without a break, therefore the need to use classification directly on the device. In addition, the use of a GPS receiver increases the functionality of the device, allowing the location of the patient to be indicated during the alarm, and allowing the patient to be found on demand.

The classification from the field of the ECG is known mainly from works involving the diagnosis of arrhythmias, which refers to cardiac arrhythmias in general. In contrast, the motivation for most researchers are incidents of myocardial infarction. Their early detection helps preventing complications such as an increased risk of a stroke or even sudden death. Analyses of the models used for this purpose over the past few years have shown different approaches, with results at different levels. For the vast majority, studies were implemented using the MIT-BIH Arrhythmia Database and the PTB Diagnostic ECG Database. The classification was most often two or five classes of arrhythmia. The assessment metric was most often Accuracy (ACC), Recall, Precision, and the F1 parameter. ACC and F1 were most often highlighted as significant. Due to the unbalanced nature of the dataset, it was often decided to introduce the Balanced Accuracy (BAcc) metric, relevant to the PTB-XL dataset.

The proposed approach shows promising results of the ECG signal classification using Tree-based ML techniques and artificial neural networks. Comparing optimizing ML models with Ref 4 (using ML models on default options), there is an improvement from optimization. It can be observed (tables 9–11) that learning_rate and n_estimators parameters are of great importance for the effectiveness of optimization. In contrast, parameters colsample_bytree and subsample are of negligible importance. Interestingly, after optimization, the LightGBM

model turned out to be significantly better, and without optimization, the XGBoost model. Although Neural Networks did not yield the expected results. This signalizes the need to carry out research that takes into account various parameters, defining classifier models, so that it would be possible to develop the best model in terms of its effectiveness and efficiency.

As it can be observed (tables 12–14), the difference between Accuracy and Balanced Accuracy metrics increases as the number of disease classes increases. For the classifications of 2 classes, the values are almost identical, while for the classifications of 15 classes, the difference is dramatic. This is because the number of elements in the classes varies – the data is unbalanced. This causes the models to learn to classify more numerous classes better because they have more such records. A likely solution to this problem would be to train models with the same number of elements in each class.

It has been proven that the proposed solution can run in real time on the IoT devices. As it can be seen in Results Section, the total analysis time of the 3-lead ECG signal (on the ESP32-S3 microprocessor) is shorter by the duration of this signal. This means that the proposed solution can run in real time on the IoT devices. For a 12-lead ECG signal, the analysis time is longer than the duration of the signal. In this case, for the device to run in real time, it is required to skip some QRS complexes.

The conducted research confirms that considering the issue of the ECG signal classification is complex and requires further research. Although the test results have shown satisfactory results, there are still some imperfections, limitations, and uncertainties, both in relation to the biological evaluation of the ECG signal (in the context of the supplied ECG waveforms within publicly available data sets) and the selection of processing and analysis methods. However, it seems indisputable that further research is needed for the sake of the relevance of the topic, which takes into account the high morbidity and mortality of heart diseases, as well as the development of machine learning methods.

References

1. Al Rahhal M.M., Bazi Y., AlHichri H., Alajlan N., Melgani F., Yager R.R., *Deep learning approach for active classification of electrocardiogram signals*. "Information Sciences", Vol. 345, 2016, 340–354, DOI: 10.1016/j.ins.2016.01.082.
2. Benjamin E.J., Muntner P., Alonso A., Bittencourt M.S., Callaway C.W., Carson A.P., et al., *Heart disease and stroke statistics—2019 update: a report from the American Heart Association*, "Circulation", Vol. 139, No. 10, 2019, e56–e528, DOI: 10.1161/CIR.0000000000000659.
3. Goldberger A.L., Amaral L.A.N., Glass L., Hausdorff J.M., Ivanov P.C., Mark R.G., Mietus J.E., Moody G.B., Peng C.K., Stanley H.E. *PhysioBank, PhysioToolkit, and Physionet: Components of a new research resource for complex physiologic signals*, "Circulation", 2000, DOI: 10.1161/01.cir.101.23.e215.
4. Hong S., Zhou Y., Shang J., Xiao C., Sun J., *Opportunities and challenges of deep learning methods for electrocardiogram data: A systematic review*. "Computers in Biology and Medicine", Vol. 122, 2020, DOI: 10.1016/j.compbiomed.2020.103801.
5. Karthiga M., Santhi V., Sountharrajan S., *Hybrid optimized convolutional neural network for efficient classification of ECG signals in healthcare monitoring*. "Biomedical Signal Processing and Control", Vol. 76, 2022, DOI: 10.1016/j.bspc.2022.103731.
6. Kiranyaz S., Ince T., Gabbouj M., *Real-time patient-specific ECG classification by 1-D convolutional neural networks*. "IEEE Transactions on Biomedical Engineering", Vol. 63, No. 3, 2016, 664–675, DOI: 10.1109/TBME.2015.2468589.
7. Korürek M., Doğan B., *ECG beat classification using particle swarm optimization and radial basis function neural network*. "Expert Systems with Applications", Vol. 37, No. 12, 2010, 7563–7569, DOI: 10.1016/j.eswa.2010.04.087.
8. Lavric A., Popa V., *LoRa wide-area networks from an Internet of Things perspective*, 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2017, DOI: 10.1109/ECAI.2017.8166397.
9. Mahesh V., Kandaswamy A., Vimal C., Sathish B., *ECG arrhythmia classification based on logistic model tree*. "Journal of Biomedical Science and Engineering", Vol. 2, No. 6, 2009, 405–411, DOI: 10.4236/jbise.2009.26058.
10. Moavenian M., Khorrami H., *A qualitative comparison of artificial neural networks and support vector machines in ECG arrhythmias classification*. "Expert Systems with Applications", Vol. 37, No. 4, 2010, 3088–3093, DOI: 10.1016/j.eswa.2009.09.021.
11. Owis M.I., Abou-Zied A.H., Youssef A.B., Kadah Y.M., *Study of features based on nonlinear dynamical modeling in ECG arrhythmia detection and classification*. "IEEE Transactions on Biomedical Engineering", Vol. 49, No. 7, 2002, 733–736, DOI: 10.1109/TBME.2002.1010858.
12. Özbay Y., Ceylan R., Karlik B., *A fuzzy clustering neural network architecture for classification of ECG arrhythmias*. "Computers in Biology and Medicine", Vol. 36, No. 4, 2006, 376–388, DOI: 10.1016/j.compbiomed.2005.01.006.
13. Özbay Y., Tezel G., *A new method for classification of ECG arrhythmias using neural network with adaptive activation function*. "Digital Signal Processing", Vol. 20, No. 4, 2010, 1040–1049, DOI: 10.1016/j.dsp.2009.10.016.
14. Pałczyński K., Śmigiel S., Ledziński D., Bujnowski S., *Study of the Few-Shot Learning for ECG Classification Based on the PTB-XL Dataset*. "Sensors", Vol. 22, No. 3, 2022, DOI: 10.3390/s22030904.
15. Perles A., Pérez-Marín E., Mercado R., Segrelles J.D., Blanquer I., Zarzo M., GarciaDiego F.J., *An energy-efficient internet of things (IoT) architecture for preventive conservation of cultural heritage*. "Future Generation Computer Systems". Vol. 81, 2018, 566–581, DOI: 10.1016/j.future.2017.06.030.
16. Roonizi E.K., Sassi R., *A signal decomposition model-based Bayesian framework for ECG components separation*. "IEEE Transactions on Signal Processing", Vol. 64, No. 3, 2015, 665–674, DOI: 10.1109/TSP.2015.2489598.
17. Sana F., Isselbacher E.M., Singh J.P., Heist E.K., Pathik B., Armoundas A.A., *Wearable devices for ambulatory cardiac monitoring: JACC state-of-the-art review*. "Journal of the American College of Cardiology". Vol. 75, No. 13, 2020, 1582–1592, DOI: 10.1016/j.jacc.2020.01.046.
18. Shadmand S., Mashoufi B., *A new personalized ECG signal classification algorithm using Block-based Neural Network and Particle Swarm Optimization*. "Biomedical Signal Processing and Control", Vol. 25, 2016, 12–23, DOI: 10.1016/j.bspc.2015.10.008.
19. Main page LoRa Alliance [on-line], accessed on: https://lora-alliance.org/
20. Śmigiel S., *ECG Classification Using Orthogonal Matching Pursuit and Machine Learning*. "Sensors", Vol. 22, No. 13, 2022, DOI: 10.3390/s22134960.
21. Śmigiel S., Pałczyński K., Ledziński D., *ECG signal classification using deep learning techniques based on the PTB-XL dataset*. "Entropy", Vol. 23, No. 9, 2021, DOI: 10.3390/e23091121.
22. Śmigiel S., Pałczyński K., Ledziński D., *Deep Learning Techniques in the Classification of ECG Signals Using*

R-Peak Detection Based on the PTB-XL Dataset. "Sensors", Vol. 21, No. 24, 2021, DOI: 10.3390/s21248174.
23. Übeyli E.D., *Combining recurrent neural networks with eigenvector methods for classification of ECG beats*. "Digital Signal Processing", Vol. 19, No. 2, 2009, 320–329, DOI: 10.1016/j.dsp.2008.09.002.
24. Wagner P., Strodthoff N., Bousseljot R.D., Kreiseler D., Lunze F.I., Samek W., Schaeffter T., *PTB-XL, a large publicly available electrocardiography dataset*. "Scientific Data", 2020, DOI: 10.1038/s41597-020-0495-6.

System do monitorowania sygnału EKG z wykorzystaniem metod uczenia maszynowego i technologii LoRa

Streszczenie: Elektrokardiogram (EKG) jest pierwszym krokiem w diagnozowaniu chorób serca. Zaburzenia rytmu serca są jednymi z wczesnych objawów chorób serca, które mogą przyczynić się do zawału serca, udaru mózgu lub nagłej śmierci pacjenta. Znaczenie EKG wzrosło wraz z rozwojem technologii opartych na uczeniu maszynowym i zdalnym monitorowaniu parametrów życiowych. W szczególności wczesne wykrywanie arytmii ma ogromne znaczenie, jeśli chodzi o diagnozowanie pacjenta z chorobą serca. Jest to możliwe dzięki rozpoznawaniu i klasyfikowaniu patologicznych wzorców w sygnale EKG. W artykule przedstawiono system zdalnego monitorowania sygnałów EKG wzbogacony o badania eksperymentalne nad zastosowaniem modeli uczenia maszynowego (ML) z grupy opartych na drzewach i architekturze sieci neuronowych, w kontekście klasyfikacji chorób serca. Badania przeprowadzono z wykorzystaniem publicznie dostępnej bazy danych sygnałów EKG, tj. PTB-XL. Wyniki analizowano w kontekście dokładności klasyfikacji dla 2, 5 i 15 klas chorób serca. Nowością w pracy jest wskazanie modeli uczenia maszynowego i architektury sieci neuronowych, jakie można zastosować w urządzeniach IoT. W oparciu o przeprowadzone badania udowodniono, że proponowane rozwiązanie może działać w czasie rzeczywistym na urządzeniach IoT.

Słowa kluczowe: urządzenie mobilne, sygnał EKG, klasyfikacja, IoT, LoRa, sieć neuronowa, uczenie maszynowe, baza danych PTB-XL

Sandra Śmigiel, PhD
sandra.smigiel@pbs.edu.pl
ORCID: 0000-0003-2459-5494

Research associate at Bydgoszcz University of Science and Technology. Ph.D. in technical sciences. Research interests include bio-signal analysis (especially ECG signal analysis), medical image processing, computer simulations in biomedicine, Machine Learning, aspects of telemedicine, telemonitoring. Co-creator of inventions in the area of developing wireless monitoring of vital functions by means of an ECG signal and software for ECG analysis.

Damian Ledziński, PhD
damian.ledzinski@pbs.edu.pl
ORCID: 0000-0003-0796-4390

Research associate at Bydgoszcz University of Science and Technology. Ph.D. in technical sciences. Research interests: software engineering, data science, deep neural networks, unmanned aerial vehicles, drone autonomy, biomedical engineering. Passionate about artificial intelligence and new technological solutions. Co-author of numerous scientific publications. Java and python developer. Popularizer of science among young people.

Prof. Tomasz Andrysiak, DSc PhD
tomasz.andrysiak@pbs.edu.pl
ORCID: 0000-0001-8138-6619

Research associate at Bydgoszcz University of Science and Technology. Ph.D. in technical sciences. Research interests: signal processing and analysis, using computational intelligence, use of methods and techniques for anomaly detection, network security issues. Expert in artificial intelligence algorithms and software engineering. Initiator of many projects. Has extensive experience in R&D projects.

Prof. Tomasz Topoliński, DSc PhD
ibm.topol@gmail.com
ORCID: 0000-0002-6232-2692

Professor and doctor of technical sciences, mechanical engineer. Research interests: engineering biomechanics (gait analysis and load distribution of human load-bearing structures), medical biomechanics (design of implants, prostheses, and orthopedic and rehabilitation devices), evaluation of mechanical and strength properties of bone structures, biomaterials, and synthetic biomedical materials. Developer of phenomenological methods for fatigue life calculations.