IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON DEEP LEARNING ALGORITHMS FOR INTERNET OF MEDICAL THINGS

Received May 28, 2020, accepted June 1, 2020, date of publication June 8, 2020, date of current version June 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3000636

Framework for Integration Decentralized and Untrusted Multi-Vendor IoMT Environments

ANDRZEJ SOBECKI[1], JULIAN SZYMAŃSKI[1], DAVID GIL[2], AND HIGINIO MORA[2], (Member, IEEE)

[1]Department of Electronic, Telecommunication and Informatics, Gdansk University of Technology, 80233 Gdansk, Poland
[2]Department of Computer Science Technology and Computation, University of Alicante, 03690 Alacante, Spain

Corresponding author: Andrzej Sobecki (andrzej.sobecki@pg.edu.pl)

This work was supported in part by the Department of Computer Architecture, Gdańsk University of Technology, in part by the Spanish Research Agency (AEI) and the European Regional Development Fund (ERDF) under Project CloudDriver4Industry TIN2017-89266-R, and in part by Grant RTI2018-094283-B-C32, ECLIPSE-UA (Spanish Ministry of Education and Science).

ABSTRACT Lack of standardization is highly visible while we use historical data sets or compare our model with others that use IoMT devices from different vendors. The problem also concerns the trust in highly decentralized and anonymous environments where sensitive data are transferred through the Internet and then are analyzed by third-party companies. In our research we propose a standard that has been implemented in the form of framework that allows describing requirements for methods and platforms that collect, manage, share, and perform data analysis form the Internet of Medical Things in order to increase trust. Further, we can distinguish two types of IoMT devices: passive and active. Passive devices measure some parameters of the body and save them in databases. Active devices have the functionality of passive devices and moreover, they can act in a defined way, eg.: inject directly into the patient's body some elements such as a medicament, electric signals to the nervous system, stimulus pacemaker, etc. Nevertheless how to create a safe and transparent environment for using data active sensors, developing safe ML models, performing medical decisions based on the created models and finally deploy this decision to the specified device. While the IoMT devices are used in real-life, professional healthcare the control system should offer tools for backtracking decisions, allowing e.g. to find who made a mistake, or which event caused a particular decision. Our framework provides backtracking in the IoMT environment in which for each medical decision supported by ML models we can prove which sensor sends the data, which data was used to create prediction/recommendation, what prediction was produced, who and when use it, what medical decision was made by who. We propose a vendor transparency framework for each IoMT devices and ML models that will process the medical data in order to increase patient's privacy and prevent for eventual data leaking.

INDEX TERMS Data vendor transparency, healthcare data analysis, IoMT fraud prevention, isolated AI algorithms, machine learning, medical decisions backtracking.

I. INTRODUCTION

The rapid development of a new type of IoMT devices dedicated to professional and non-professional customers conduct a problem with vendor lock-down, ease integration, validation different data sources, and finally securely management of the created data sets which permissions who can use the data. Many of these devices require dedicated applications that prevent automatic analysis in real-time using your own machine learning models, e.g. [1]. The problem is more complex when we need to analyze data from different sensors at once or we have to monitor and compare in real-time more people using some set of IoT devices from different vendors. The problem of integration data exists also after they are stored in the hospital data centers because data are fragmented, formats are not standardized and platforms do not offer access to data. Moreover, the methods which we have to use in order to search and obtain the data from these data centers also differ. This is crucial for training and adjusting ML models because they need historical data sets. The second problem concern reliability and trustworthiness of IoMT solutions such as sensors, wearable devices, infrastructure, ml models, data sets, and decision repositories, or collaborative platforms. Until we used the IoMT for non-professional

The associate editor coordinating the review of this manuscript and approving it for publication was Marcin Woźniak.

purposes then reliability and trustworthiness are not so crucial as when we have to use these devices in order to save our life. Currently, the IoMT is a bunch of devices in which many are similar to the prototype created in few hours based on components available in the market such as Arduino, Rasberry Pi, Bluetooth, or WiFi module, chemical or physical sensors, etc. Thus devices we called IoMT because they can measure something in our body and can send measured data through the network or save them in the internal memory. In our perspective adjective ''medical'' needs to point out only thus devices that meet a set of requirements that will ensure a certain level of reliability and accuracy. Moreover, the software for IoMT should respect some well-defined interface in order to increase transparency which will provide to reduce or eliminate security problems and data leaks. The transparency will be also valuable while we want to integrate different devices and software vendors, medical experts, ML vendors, and give them access to a huge amount of data. Who should get access and who shouldn't, who should give permission, and for how long. Thus questions will be more important while most of the people will have the home IoMT kit to prepare themselves or remote diagnosing. Finally, transparency is crucial if we want to control and prevent personal data trading.

The contribution of this paper is a proposition of standardization the process of read and usage data from IoMT sensors to prepare professional Data Broker Platforms. We propose also a new model of developing AI algorithms and ML models that prevent fraud and data leaks. We assumed that data from IoMT should be considered as Protected Health Information (PHI) and that require special Data Broker Platform with possibility to training, testing and evaluating algorithms using the isolated containers without Internet connection. We propose to organize Data Broker Platform in decentralized manner to increase availability of data. Finally based on the performed experiments we prove that the proposed solutions work and enable easily access to the data and results through the Internet and dedicated desktop application.

The paper is organized as follows. Section II describes actual solutions available in the market in particular the IoMT devices (see Section II-A) and platforms for data processing and collection (see Section II-B). Section III introduces basic constraints used throughout this paper. Sections IV to VI describe layers of the proposed framework: data acquisition (Edge layer), data brokers (Middle layer) and data analytic (User layer). Section VII contains description of performed tests and the experimental environment. Finally, Section VIII presents our conclusions.

II. STATE OF THE ART

A. IoMT SOLUTIONS

One of the popular devices is AIO Sleeve 2.0 [2] which is a sleeve with electrodes, battery, and data transmitter that sending data to a dedicated mobile application. The efficiency of this solution is very good but the set is very sensitive to the distance between the phone and the data transmitter (BLE 4.0). The device uses the photoplethysmography [3] method to monitor heart rate. The AIO Chip has integrated high precision accelerometer which provides to capture the metabolic equivalent of task (MET) [4]. We can not easily perform custom on-line analysis of our data. There is exist an application for mobile devices and online portal offered by the company KOMODO.

Consensys Development Kit [5] is the first professional device kit that provides us to combine which metrics we want to measure and how we will process the data. From Consensys we can get full API with documentation that allows us to create a custom application and send data from sensors everywhere in RAW format. In the development Kit we get a dock station Base15 and some subset of sensors such as IMU (Internal Measurement Unit), ECG, EMG (Electromyography), and GSR (Galvanic Skin Response).

Polar H10 [6] is a belt that offers the possibility of monitoring heart rate and is compatible with some non-professional applications available on the market such as Endomondo or Runtastic. The device has better support from the dedicated mobile application. We can connect the device with other sensors using ANT+ technology. Like most other products available in the market we can not send data in real-time to analysis to the selected Data Broker Platform. It is possible to register data in the device and copy them after the end of the measurement. There is a lack of information about the possibility to export data to other platforms.

Hexoskin Pro [7] is a device that allows for monitoring HRV (Heart Rate Variability) [8]. The set contains sensors, clothing, mobile, and desktop applications for management devices and web portal to analyze data. Currently, it is not available to analyze data in real-time in the provided web portal or send data to other Data Broker Platform. The user should after training connect the device to the computer and manually transfer the data. The data format is proprietary by the manufacturer.

Garmin Forerunner 945 [9] is a smartwatch device that allows monitoring HRV, temperature, localization, wrist pulse, the degree of acclimatization (pulsoxymetr). The device can communicate through Wi-Fi, Bluetooth, and ANT+. The device offers a set of services for processing data stored in the smartwatch and offers results in well-known metrics. We can not define the target where we want to send data. This is an example of a vendor lock-in solution.

RESPA Elite Fitness [10] is a device to monitoring respiratory rate which means how many breaths are made per minute. This device calculates the metric based on signals registered by the microphone which should be installed close to the neck. The Respa has alarm functionality which is automatically launched when the respiratory frequency limits for the selected activity are exceeded. We can use a dedicated mobile application to observe the results generated by the device. Only in the Coach version, we have the possibility to remote analysis data in the near real-time.

Kenzen Patch [11] is a device that helps prevent dehydration through monitoring skin perspiration and temperature. It is installed in the form of a slice glued to the chest or arm.

TABLE 1. Summary of IoMT devices available in the market. Labels of devices presented in the Table: 1 - AIO Sleeve 2.0, 2 - Polar H10, 3 - Shimmer Consensys Development Kit, 4 - Hexoskin Pro, 5 - Forerunner 945, 6 - Kenzen Patch, 7 - RESPA Elite Fitness 3.0, 8 - Physilog 5, 9 - Smart Sock V2.0, 10 - GPEXE Pro, 11 - ClearSky, 12 - OpenBCI All-in-one Development Kit, X - presence of the feature.

Features	Devices											
	1	2	3	4	5	6	7	8	9	10	11	12
Heart rate	X	X	X	X	X	X				X	X	X
Heart rate variability	X			X	X							X
Breathing frequency				X	X		X					
Skin sweating						X						X
Gait quantitative analysis	X			X	X			X	X			
Gait qualitative analysis								X	X	X		
Speed and distance		X		X				X	X	X	X	
Location in an open space		X			X				X	X		
Location in an closed space											X	
Mobile application for online monitoring	X	X	X	X	X	X			X	X	X	
Web application for online monitoring											X	
Monitoring multiple people at once			X							X	X	
Available for individual customers	X	X	X	X	X	X		X	X			X

Using the dedicated mobile application it is possible to view in the real-time metric values generated based on the signals from the device. We can use also application dedicated to monitoring a group of users such as Kenzen Team Dashboard.

GaitUp Physilog 5 [12] is a small sensor that can be applied by installing it at the top of the shoe. It should be integrated with a dedicated mobile application or desktop application. Using the application we can analyze a running technique under all conditions or help with adjusting shoes. The application offers additional metrics such as the foot contact time, step time, speed of running, asymmetry of steps, foot angle of contact with the ground, force of pressure, and many others. It is not possible to process data in real-time. We should wait until we end the training and then we can copy or analyze data stored in the device.

Smart Sock V2.0 [13] is a product created by the Sensoria company. It is a device in the form of a sock that embedded sensors created from textiles and the inertial measurement unit called Sensoria Core (Bluetooth transmitter). The solution offers information about a number of steps, foot contact time, pressure, and running pace. We can use the mobile application which presents in the real-time values of metrics or we can use the internet application to review the historical data.

GPEXE PRO [14] is a wearable solution to precise tracking the movement of the user. The product is created by s device installed in a special t-shirt. To monitor the position of user we can use internet applications or mobile apps dedicated to iPad devices. Only on the iPad, we could view data in real-time while the desktop app provides to send data to the Internet application. We can not export data to some standardized format or register them in the other Data Broker Platform.

Catapult ClearSky [15] offers the possibility of measuring the position and velocity of the user. Also, the device is dedicated to using in the closed area because it uses a UWB (Ultra-wideband) [16] technology which allows determining the player's position relative to the receivers (called anchors) on the pitch. We can use a dedicated mobile and web application that provide access to data in real-time.

OpenBCI All-in-one Biosensing [17] is the second professional device kit that provides us to use the RAW data and processing them using our custom algorithms. It is also a device in which we can integrate different sensors to better monitoring and understand changes in the user's body. All-in-One device kit contains set with Headband kit to EEG monitoring, ECG electrodes, and Muscle Sensor. All sensors should be connected to the Biosensing Board and then we can use dedicated software to analyze data or we can write own solution.

B. PLATFORMS AND TOOLS FOR IoMT

Apple proposes the platforms ResearchKit and CareKit [18]. They were created to collect and process healthcare records that were recorded using Apple's devices. ResearchKit [19] is an open-source framework that can be used by researchers to create applications dedicated to iPhones or Apple Watches to collect medical data or performing some medical analysis. ResearchKit uses information from the embedded Health app. There are exists some examples of applications uses ResearchKit that improve the quality of user's life for example by changing diet (GlucoSuccess app). Asthma Health app is focused on getting more data about people with Asthma and their behavior. The big disadvantage of the ResearchKit is a narrow set of potential patients reduced to the owners of Apple products. We have also lack of information about the privacy of users' data.

Microsoft creates a platform called Health Vault or eHealth Information Management [20]. They offer automated talking services called Microsoft Healthcare Bot which uses AI to perform conversation. The user can talk with the Healthcare Bot about medical services and advice. Microsoft Immunomics is a project which allows diagnosing disease based on the patient's antigen map. The project Antigen Map concerns building a complete mapping of T-cells to antigens. Microsoft also promotes the Ambient Clinical Intelligence Solutions which assumes that the doctor is focusing only on patients and clinical documentation writes itself. In this project, they cooperate with Nuance DAX [21]. This functionality uses algorithms for speech recognition, intelligent

TABLE 2. Platforms that support collecting data from IoMT.

Name	Architecture	Data sources	Export format	Data privacy	Integrated ML Support	User privacy
Apple ResearchKit	On-device (distributed)	HealthKit	Custom	Permission given once by patient (data are downloaded company)	No	Platform depends
Microsoft eHealth Information Management	Cloud	API for FHIR,	FHIR	Permission given once by patient (data are downloaded company)	On the Azure Platform	Platform depends
1up.health	Cloud	FHIR, Fitbit, HL7, 5833 EHR systems,	FHIR	Hospital is owner of data	No	Platform depends
onFHIR.io	On-Premise/Cloud	FHIR	FHIR	Permission given once	No	Platform depends
Bridgera Monitoring	Cloud	Bridgera myHealth	Custom	Permission given once	No	Platform depends
DocBoxMed	Lack of information	Custom set of IoMT devices	Custom	Hospital decide	Lack of information	Platform depends

translation, and summarization. All notes and documents are protected using voice biometric authentication. Finally the Microsoft offer through Azure platform API for FHIR (Fast Healthcare Interoperability Resource) [22]. It is a solution for the easy integration of electronic medical records using the HL7 FHIR standard. They offer also a possibility to share the data between units registered in a virtual organization. Microsoft creates also the IoMT FHIR Connector for Azure which is an open-source project [23] for collecting data from IoMT devices and store them in the Microsoft FHIR Server. The software to create a server is also available in the GitHub like an open-source project [24].

1up.health [25] is a platform for sharing medical data and integrating data sources such as EHR systems Epic, Cerner, Allscripts, Meditech, eClinicalWorks, Athenahealth, Aprima, SRS Health, Greenway, NextGen Healthcare, PointClickCare. The platform can also be integrated with medical devices through 1upHealth API. Data collected on the platform are available for hospitals, patients, and developers.

The next solution available in the market is onFHIR.io [26] which enables the possibility to create its own repository of Electronic Health Records. The integration process supports a range of protocols like IHE Cross Enterprise Document Reliable XDR, KAFKA, CCD/CDA, HealthKit, FitBit API, Open mHealthAPI, and Google Fit. Most of the protocols are available only in the Enterprise version of the software. Data collected in the onFHIR platform could be processed in the dedicated cluster with Apache Spark and MongoDB software.

Bridgera Monitoring [27] is a solution dedicated to remote monitoring patients in real-time. Data from devices may be observed on the dedicated web platform or mobile application. Developers can create software and get access to data collected from devices through standard protocols such as MQTT or TCP. The platform doesn't offer the possibility to develop AI/ML solution and uses classical data formats such as JSON, CSV. Based on the collected data we can observe trends and configure custom alerts.

DocBoxMed [28] is a solution dedicated to integrate devices used in hospitals to continuously monitor patients on Intensive Care Units. They propose to integrate medical devices with the platform to perform automatic analysis performed by the AI/ML models created by the healthcare providers. Based on the collected information the DocBoxMed platform offers real-time metrics and can send alerts to the medical staff caring for the patient.

Most of the presented devices and all of the presented platforms propose solutions that required from us deciding in which company we should store our data. But that creates problems which trust to 3rd party vendors of software which can easily copy and process data even after patient withdraw consent to their processing. The existing platforms centralized data collecting them from different vendors and then can share data using a standard like a HL7 FHIR. Currently, the platforms (see Table 2) seems to have limited support for real-time analysis and usage of the metrics' values in order to improve our life, e.g. by creating more intelligent home assistants [29]. In this paper, we focus on a framework that is dedicated to collect data from many IoMT devices in real-time and to perform AI/ML analysis in privacy-be-design isolated containers. Metrics generated in our framework can be easily used in other software through dedicated API and patients' data does not leak from the platform.

III. PROPOSED FRAMEWORK

The proposed framework (see Fig. 1) assumes that it is possible to mitigate potential drawbacks to create secure platforms for aggregating, brokering, and performing medical data with full backtracking. The secure and transparent platform for professional analyzing data from IoMT should offer:

- signed sensor data – each medical fact should be automatically signed by the device and the monitored person;
- open for ML models – testing and deployment of ML models should be supported by the data broker platform;
- prevention of leaks – ML models should be executed in an isolated environment and have access to data through the specified API;
- validated ML models – each ML model accepted by developers will be automatically validated by the

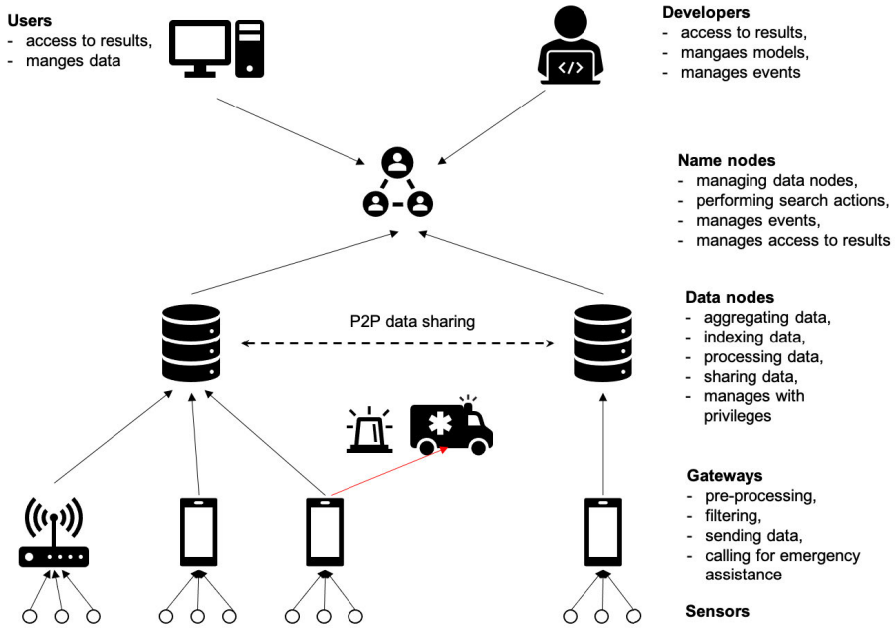

FIGURE 1. Architecture of the proposed solution.

platform through unified API using medical records unknown for the developers;
- registered data usage – each data access by the ML model should be registered;
- signed recommendations – recommendations created by validated models should be registered with information about the version of the model, used records;
- signed medical decision – decision made by a doctor should be also registered with additional information (generated recommendations, used ML models, used data, etc.);
- decision backtracking – based on the registered data we should have the possibility to describe each decision as a sequence of steps from a doctor/ML model back to the source of medical data on which based is;
- avoid lock-in – platform should be vendor-agnostic which means that we can store data from different devices regardless of manufacturer and the user should have a possibility to easy export and delete data using the unified format accepted by other, similar platforms;
- management of data – the user is the owner of the data and is free to determine to whom and for what purpose it will be made available.

In our research, we propose a model of data from IoMT in which we can connect data with the decision. This creates an opportunity to discover new types of relations for example how drugs influence some health metrics. Thus information may be useful for pharmaceuticals companies that may be interested in better testing their drugs. If we can backtracking all decisions then we have the possibility to easier detecting some anomalies or pandemics by comparing current results with historical data. In the proposed framework backtracking of medical decisions means that we can divide responsibility for erroneous decisions. Usual the EHR platforms collect data, process them with anonymous algorithms, and give for the doctor results, advice, recommendations. The doctor prepare decision based on the results generated by an anonymous algorithm. We propose in our framework track each medical decision and describe it by devices, sensors, algorithms, results which were used by the doctor while he prepares a certain decision. If the decision will be erroneous then we can check who fails e.g., which version of the algorithm, storage platform (broken data), device, or sensor. A platform that satisfies all of the mentioned above requirements eliminates or reduces the risk of locking the user and his data in some cloud platform from which he can't export them. Finally, the described standard enables a discussion on how to increase the safety usage of IoMT in professional use cases.

IV. EDGE INFRASTRUCTURE

Currently, the IoMT devices may be built from all components that are available in the market. How we can inform potential patients, athletes, or hospitals that some device is no longer safe? Today when we are at the start of the IoMT road we should prepare standards describing how the devices should be tested and which criterion they should satisfy, e.g. precision of measurement, the precision of dosage, accuracy of electromagnetic pulse control, etc. Moreover, we have to define today where and how we will publish information that some devices may have potential problems. Finally, the IoMT devices usually send and receive data through the Internet so particularly important are cyber-safety and potential software drawbacks.

We propose the following requirements for edge IoMT infrastructure:

- hardware and software components of IoMT are public – we should avoid usage IoMT devices that are based on hardware or software components described as not safe (have known vulnerability). To reduce the risk of using those components we propose to add for each IoMT device a list of components that will be publicly available.
- user safety – each IoMT device should use current cryptography solutions to encrypt and sign data received from sensors which are sent to gateways or data broker platforms.
- patient safety – we recommend to usage gateway in which data from IoMT are analyzed before they were sent to the data broker platform. This may be crucial while signals from IoMT point out that there are required emergency help. Moreover, in the gateway, we can usually install more complex software which can translate medical decision into the hardware component signals.
- vendor-agnostic – each IoMT device should offer public API for accessing to data registered by sensors in some universal format such as shown in Listing 1 where *userId* is a unique user identifier, *deviceId* is a gateway id, *sensorId* is a unique id of sensor which produce data, *typeOfSensor* is a proposed classification that unified types of sensors and make easier the process of finding appropriate data, *timestamp* is a Unix-like description of date and time when data was registered by the gateway, *sessionId* is a unique value for the user which describe that measured value comes from one training/measurement session, *objectAsBase*64 is any value obtained by the sensor *sensorId* encoded using base64, *hashValue* is a result of a hash function (e.g., SHA-256 or SHA-512) from a concatenation of all mentioned attributes (string values).

```
{
    'userId',
    'deviceId',
    'sensorId',
    'typeOfSensor',
    'timestamp',
    'sessionId',
    objectAsBase64,
    'hashValue'
}
```

Listing. 1. Example of unified dataframe from IoMT device sensor.

The edge infrastructure in the proposed framework can act as a gateway but also in a more complex scenario as a peer (data node) or even a compute node. This will ensure a high level of privacy and provide us to create applications in fog computing architecture in which we use for example deep learning's capabilities of edge devices [30]. Moreover, the smartphone may receive feedback from the AI/ML models run on the server or smartphone, and based on it perform some actions such as adjust the operation of the insulin pump or pacemaker.

V. AGGREGATING, BROKERING AND PROCESSING DATA

The patient should have information on what data he produce, where they are collected, who use them, and should have the option to gain access to results of analysis based on his data. The platform that integrates data from many users and sensors should be transparent for data providers (patents, athletes, etc.). Lack of transparency may provide to data leakage, vendor lock-in, and usage of models with unconfirmed effectiveness. In the proposed platform we should also accept data from Apple ResearchKit or professional HIS using the corporate standard HL7 FHIR [31].

Currently, we can observe issues when we try to use at the same time wearable devices developed by different vendors such as Apple, Samsung, Garmin, AIO Sleeve, Hexoskin. They offer dedicated applications with some subset of metrics with unknown precision and effectiveness. Data stored in thus applications may in some conditions be exported after we finish the monitoring process. The biggest problem occurs when we have to monitor in real-time user or group of users who use many sensors from different vendors.

Data from IoMT may be sensitive and we should care for them like a PHI (protected health information) which means that platforms should comply with existing standards such as HIPAA. But due to the type of this data, they need usually to be analyzed using AI or general ML models before they will be presented to a doctor. We propose to define additional requirements for the IoMT data broker platform to ensure easier access and management of data by the patient, and standardized access to the data by ML models. The proposed standard was described in the section VI.

Data brokering services should be better organized to enable migrating data between providers, prevent data leaks, and reduce uncontrolled processing data without significant benefits for the owner of the data. Naturally, the data from IoMT will be aggregated in multiple data brokering platforms due to localization of the user, types of used devices, access to better ML models, etc. It is not reasonable to force creating a single point in which all data should be stored because this usually provides synchronization and efficiency problems. Moreover, it is not secure if all data will be stored in one organization due to DDoS attacks for example, or simple failure of infrastructure.

Currently, most platforms work in the cloud which provides to the centralization of medical data. We propose a decentralized environment based on the P2P network in which each of the peers is a single data brokering platform with some parts of the user's data. Peers may be localized and provided by hospitals, government units, universities, or non-profit organizations. To create a simple P2P cluster for data we can use the IPFS-cluster (Inter-Planetary File System) software. Selecting the peer in the proposed model is depends on user preferences, type of device, his localization, and his doctor. The user may decide if he wants to create

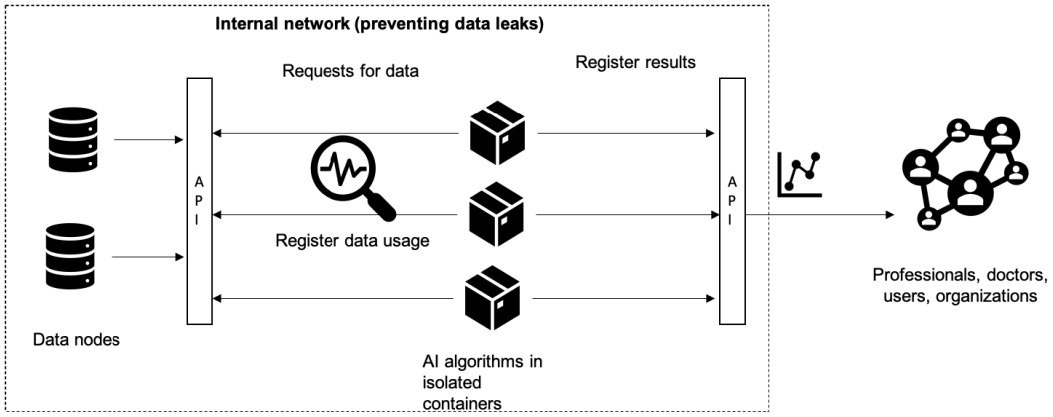

FIGURE 2. The proposed architecture of isolated containers.

public data sets accessible for everyone or does the measured data are sensitive and should be confidential and available only for doctors. The proposed P2P architecture makes easier parallel processing data from people distributed around the world. P2P architecture ensures also the possibility to access data that are available in other nodes. In the presented version we don't provide assumptions about a secure privileges management functionality which should share in decentralized manner information about who can access which data sets.

We assumed that the user has a single and unique ID number in each of the node and each node have a unique ID number. Based on the user's ID it is possible to find all data created by the same user. To better organize the process of searching data from a specified user we propose to add special nodes (seed node) that contain only information about available data nodes and users' IDs which have data on the specified node. Each data node should communicate with one of the seed nodes in some defined time. Seed nodes communicate with each other to broadcasting keep-alive information. The user application communicates with one of the known seed nodes while he starts the search process. The requested seed node check if he has some data from user's and parallel ask other seed nodes about data belongs to the specified user. In response, the user's application will get a list of stored data sets grouped by the type of sensor (typeOfSensor).

Organizations may also search for data but before they can access the data they have to get permission from the user. Depends on the selected data set the data node who stores the data set will send a request to the user and wait for a response in the application. We assumed that each data sharing should be realized in something called an event. Each event should be described by name, founder ID (ID of a registered organization), start time, end time, description of the purpose, do results will be available for the user. The user's decision about sharing data only applies to the specific event.

The HL7 FHIR is a standard of the description of the Electronic Health Records and the proposed framework can use this standard to store data. Both HL7 FHIR and the proposed framework provide unification medical data but in our proposition, we focused only on the frequently updated data from IoMT. Moreover, the proposed framework assumes that data stored in the servers are not available for download but can by only used through dedicated API available for isolated containers. In our framework, the patient is the owner of data. We provide a possibility to compare the efficiency and accuracy of ML models that were prepared based on data stored in the platform. Our framework describes the mechanism of ensuring trust in a decentralized environment in which we use models created by different vendors. HL7 FHIR does not comply with these requirements. HL7 FHIR does not specify the aspect of using the IoMT or ML models.

VI. TRUSTED AI ALGORITHMS AND ML MODELS

Usage of AI and ML solutions in the analysis based on IoMT required standardized and secure access to data stored by different data brokers and universal methods to transparent validation thus algorithms. Who should get access to data and who should manage with thus privileges? We propose to use the isolated containers (see Fig. 2) in which each AI algorithm could be trained, tested, and finally evaluated. We assume that each data broker platform will offer set or resources that are available for IT developers who want to create an AI/ML solution. Through an isolated container, we ensure the security of data while the model in the container can't send any data to the Internet. The current version of the framework does not include prioritization functionality. Data from each sensor are processed in the same manner. In the future this should be changed in order to reserve resources for certain models.

The second advantage of the proposed solution is full control of which data was used by the algorithm and which accuracy it offers. While developers want to deploy their model into the platform we also may have information which libraries in which version are currently used. We propose to use the same standard of API in all data brokers platform which provides easier migrating models generated in different platforms. While we have knowledge about which data

was used by which algorithm we can easily backtracking each decision. We propose to use the performance testing and evaluated using the same API interface that will be used to generate results in the final environment. This will solve also one of the biggest problems with new technology in healthcare that doctors should use different programs to get the results of data analysis. We propose to unify this by data broker platform in which the AI algorithm is executed in an isolated environment.

Currently platforms usually share patient's data after he gives permission. The sharing process is realized by creating some endpoints from which the destination application can download a copy of that data. If the application is created and maintained by the patient's hospital, then the patient may in some context trust them. But more often application is prepared by 3rd party companies that after download data may use them without any additional permissions of the patient. Theoretically data should be anonymized, but which elements are removed or obfuscated depends only on the platform. To increase the level of data and user privacy we propose to share data only through dedicated API which is accessible from a specially isolated environment. Input and Output operations may be performed only by using the mentioned API, and the application has no connection to the resources outside the isolated environment. In that manner we ensure that patient's data was not downloaded by any 3rd party company. In the proposed model patient is the owner of data not a hospital as usual.

The third advantage concern the ability to compare AI models that are developed and publish through the data brokers platform. Currently, we should each time create some baseline algorithms and compare our model or algorithm with that baseline. Finally, when we deploy algorithms in the real environment we may achieve different results. To eliminate thus drawbacks, we propose to perform the automatic evaluation on the data broker platform when developers decide to publish and deploy the algorithm in the market.

In the proposed framework ML models can be trained, tested, and evaluated only in the isolated containers using the data through the dedicated API. We propose to divide set of available data into two subsets of data: 1) data that are accessible by the models while they are created and 2) data that may be used only by the platform to evaluate the model while he is published. Moreover, the dedicated API offers access to data and receives the results from the ML models. The proposed framework can register accuracy and efficiency achieved by the model developed in the platform. The results have a unique ID and can be published and compare later with other models.

To perform the automatic evaluation we propose to use subsets of IoMT data which:

- belongs to more than one patient,
- was not used in training or testing phases,
- have results confirmed by doctors.

Moreover, each validation process is signed by the data broker platform with information such as the evaluation date, name, and version of algorithm, reference to a subset of records used, and set metrics (e.g., F1, precision, recall, etc.). We assumed that in professional healthcare we should use only thus models that have proved their accuracy in one of the trusted data broker platforms. In that case, everyone can check the accuracy of the AI algorithm or ML model which will be provided by their authors.

VII. EXPERIMENTS

A. TEST ENVIRONMENT

The experiments were conducted in the Gdansk University of Technology using the following set of sensors:

- Shimmer3 IMU Unit,
- Shimmer3 ECG Unit,
- Shimmer3 EMG Unit,
- Shimmer3 GSR Unit,
- Garmin HRM-Tri,
- Garmin Forerunner 910XT,
- Smartband HUAWEI Band A2,
- OMRON M2.

As a baseline was used the pressure gauge OMRON M2 Basic, Smartband HUAWEI Band A2, Garmin Forerunner 910XT with HRM-Tri. We focus on integration and processing raw data from heart monitoring sensors and inertial measurement units. We checked what is an efficiency of the algorithm deployed in the proposed Data Broker Platform dedicated for IoMT.

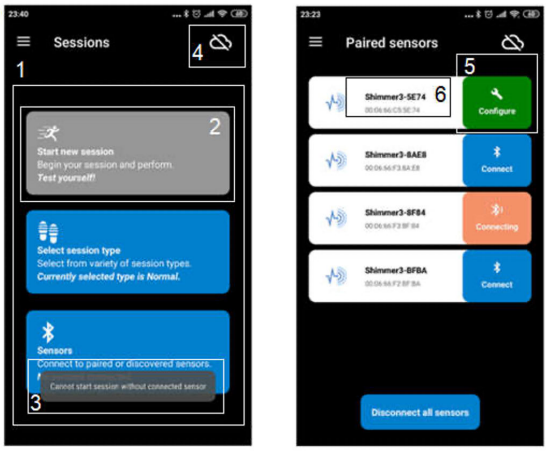

FIGURE 3. Dedicated application for gateway device.

Experiments were performed using the Smartphone Samsung Galaxy S10 with LTE as a Gateway mediating communication between sensors and the proposed Data Broker Platform. For the gateway, we prepare a dedicated application (see Fig. 3) to managed IoMT devices and pre-processing data from sensors. The platform was launched in the OpenStack private Cloud using Vhost with 2 vCores, 8GB RAM, 10GB of HDD, and 1 Gbps Internet connection. Results of the analysis were observed in the dedicated application (see Fig. 4 and Fig. 5) launched in the Macbook Pro with Core i5, 8GB RAM, 256GB SSD, and 1 Gbps Internet connection.

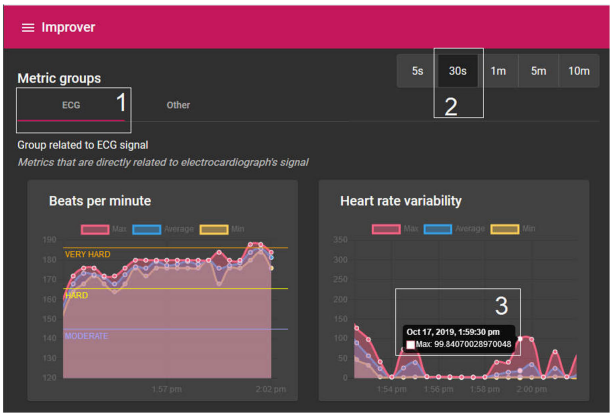

FIGURE 4. Dedicated application for managing and access to results from the Data Broker Platform.

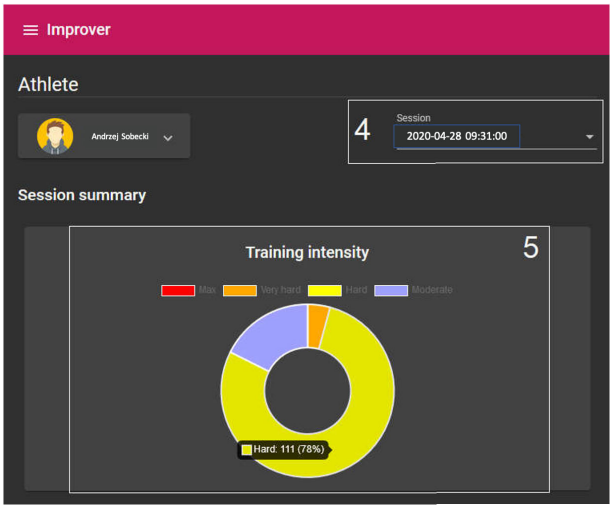

FIGURE 5. Dedicated application for managing and access to results from the Data Broker Platform.

The application was connected to the DBP through the internal network.

B. RESULTS

The first example algorithm deployed in the data broker portal conduct processing the RAW data signals and transform them into signals **R** as shown in Fig. 6.

First, in order to check the prepared algorithm, we used data set MIT-BIH Noise Stress Test Database [32], [33]. The data set contains raw signals with different SNR (signal-to-noise ratio). When the ratio is lower then the difference between noise and signal is smaller and its harder to parse data. We use two data sets with SNR 24dB and 12dB. Using these data sets we simulate the IoMT devices by sending signals through the prepared mobile gateway and next to the data broker platform. Examples of signals processed in these experiments were shown in Fig. 7 and Fig. 8.

The created algorithm detect properly most of the R-point in ECG data. For signals with SNR 24 dB mean absolute error was 3.6 (4.4%) and for SNR 12 dB MAE was 4 (4, 89%). Based on the created algorithm we prepare extension which

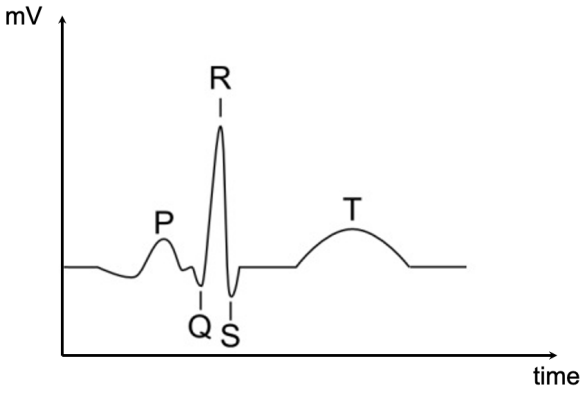

FIGURE 6. Example of signals' positions on the EKG signal.

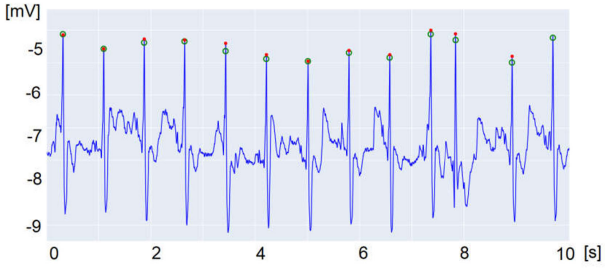

FIGURE 7. Example of EKG signals with SNR 24dB.

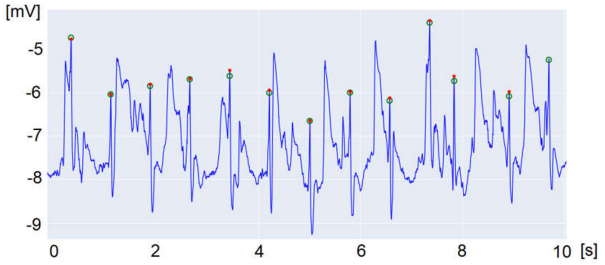

FIGURE 8. Example of EKG signals with SNR 12dB.

TABLE 3. Example of results of processing data in the data broker platform generated based on MIT-BIH Noise Stress Test Database - R-point occurrences test.

No.	Base line	SNR 24 dB	SNR 12 dB	Latency [ms]
1	87	83	83	1399
2	79	76	75	1270
3	79	75	75	1310

calculates heart rate [34] using information about R-points in the ECG signal. The results were compared to devices Smartband HUAWEI Band A2 and OMRON M2 Basic.

The next experiment concerns analysis of the heart rate using the vendor devices such as OMRON M2 and Smartband HUAWEI Band A2, and Shime Sensing ECG with our gateway and data broker platform. Using the extended version of the algorithm described above we try to compute heart rate based on data sent from the IoMT device through the gateway to the data broker platform. The algorithm was deployed in the isolated container and was called each time when new

data was registered on the platform. The user was measured three before he did the exercise, then he did the exercise for 10 minutes and finally took again a measurement. The experiment was repeated three times at 30-minute intervals and the results were averaged and showed in Table 4.

TABLE 4. Results of processing data by the DBP (data broker platform) in compare with vendor solutions.

No.	OMRON M2	Band A2	DBP
	Heart rate	Heart rate	Heart rate (Latency [ms])
1	70	74	72 (2301)
2	72	76	73 (2117)
3	81	85	82 (2090)
4	105	118	112 (1711)
5	126	130	129 (1645)
6	102	94	101 (1899)

As we can observe the proposed solution compute result with mean latency 2169 ms for a user before exercise and 1752 ms for a user after exercise. The mean average error was 3.33 BPM for a user before exercise and 3.67 BPM after the exercise. While the Smartband HUAWEI Band A2 solution achieves MAE equal 4 BPM before exercise and 8.33 BPM after exercise.

In the parallel of information from the ECG sensor, we send information about the motion of the user. Motion data was created by the Simmer3 Inertial Measurement Unit attached to the user and connected to our gateway. In the experiment, we want to check the accuracy and latency of our algorithm for counting the number of steps that were launched in the proposed DBP. We perform 10 experiments from which five experiments were about walking in a calm way and five experiments about fast running. The results of the experiments were presented in the Table 5 for a calm walk and in the Table 6 for fast running.

TABLE 5. Results of algorithm computing number of steps based on IoMT data - calm walk.

No.	Base line	Computed value (Mean latency)	Diff
1	50	50 (1901 ms)	0
2	100	102 (1843 ms)	2
3	150	152 (1799 ms)	2
4	200	204 (1795 ms)	4
5	250	252 (1793 ms)	2

TABLE 6. Results of algorithm computing number of steps based on IoMT data - fast running.

No.	Base line	Computed value (Mean latency)	Diff
1	50	52 (2211 ms)	2
2	100	100 (2303 ms)	0
3	150	152 (2120 ms)	2
4	200	204 (2112 ms)	4
5	250	252 (2233 ms)	2

The algorithm in each variant of the experiment achieves high accuracy. Maximal differences between the real and measured number of steps were 2 regardless of how the user walked. The presented latency was the result of delays in the process of data transmission from the sensor through the gateway to the platform.

VIII. CONCLUSION AND FUTURE WORKS

Usage of AI algorithms and ML models in professional healthcare requires tools and platforms that ensure high-quality data, validated algorithms, and comparable results. Moreover, thus solutions should offer easy and secure access to the unified data for patients, doctors, and organizations.

We proposed the architecture of the data broker platform which unifies data aggregated from IoMT edge devices created by different vendors and offers an isolated environment for training, testing and evaluating the AI algorithms preventing fraud and data leaks. The unified data model assumes that a mobile device (e.g. smartphone) acts as a mediator that performs pre-processing, unification of data, signing them, and registering in the selected data broker platform. Each algorithm may access data only after registration on the data broker platform in which will be executed and tested. Finally, the proposed solution offers a trusted evaluation mechanism for the algorithm and an integrated platform for validation and comparison results. After the algorithm was evaluated and achieve the minimum required values of metrics then it may be distributed publicly for trusted partners such as hospitals, universities, government organizations (courts), and private companies (insurance, pharmaceutical, sports coaches, etc.). The proposed data broker platform may act as a mediator between thus organizations, algorithms developers, and users ensuring the quality of data, algorithms, and results.

In future work, we should analyze the possibility of usage the block-chain technology in order to create a decentralized and safety mechanism of management privileges and consents. Probably only the decentralized environment protects users from losing control of their data or being dependent on a single provider. Finally, we should try to create a decentralized platform in which we can create and distribute a validated and comparable AI algorithms based on the IoMT data.

REFERENCES

[1] H. Mora, D. Gil, R. M. Terol, J. Azorín, and J. Szymanski, "An IoT-based computational framework for healthcare monitoring in mobile environments," *Sensors*, vol. 17, no. 10, p. 2302, Oct. 2017, doi: 10.3390/s17102302.

[2] *Komodotec Aio Sleeve 2.0*. Accessed: Apr. 28, 2020. [Online]. Available: https://komodotec.com

[3] J. Allen, "Photoplethysmography and its application in clinical physiological measurement," *Physiolog. Meas.*, vol. 28, no. 3, pp. R1–R39, Mar. 2007.

[4] H. van Remoortel, C. A. Camillo, D. Langer, M. Hornikx, H. Demeyer, C. Burtin, M. Decramer, R. Gosselink, W. Janssens, and T. Troosters, "Moderate intense physical activity depends on selected metabolic equivalent of task (MET) cut-off and type of data analysis," *PLoS ONE*, vol. 8, no. 12, Dec. 2013, Art. no. e84365.

[5] *Consensys Development Kit*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.shimmersensing.com/images/uploads/docs/ConsensysPRO_Spec_Sheet_v1.1.0.pdf

[6] *Polar H10*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.polar.com/en/products/accessories/H10_heart_rate_sensor

[7] *Hexoskin*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.hexoskin.com/collections/all

[8] U. R. Acharya, K. P. Joseph, N. Kannathal, C. M. Lim, and J. S. Suri, "Heart rate variability: A review," *Med. Biol. Eng. Comput.*, vol. 44, no. 12, pp. 1031–1051, Dec. 2006.

[9] *Garminforerunner*. Accessed: Apr. 28, 2020. [Online]. Available: https://buy.garmin.com/pl-PL/PL/p/621922#specs
[10] *Respa*. [Online]. Accessed: Apr. 28, 2020. Available: https://www.zansors.com/respa
[11] *Kenzen*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.kenzen.com/patch
[12] *Gaitup*. Accessed: Apr. 28, 2020. [Online]. Available: https://shop.gaitup.com/index.php?id_product=1&controller=product
[13] *Skarpety*. Accessed: Apr. 28, 2020. [Online]. Available: http://store.sensoriafitness.com/smart-sock-v2-0-sensoria-core/
[14] *Gpexe*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.gpexe.com
[15] *Catapult*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.catapultsports.com/products/clearsky-t6
[16] J. D. Taylor, *Ultra-Wideband Radar Technology*. Boca Raton, FL, USA: CRC Press, 2018.
[17] V. Peterson, C. Galván, H. Hernández, and R. Spies, "A feasibility study of a complete low-cost consumer-grade brain-computer interface system," *Heliyon*, vol. 6, no. 3, Mar. 2020, Art. no. e03425.
[18] *Researchkit*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.apple.com/pl/researchkit/
[19] *Researchkit2*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.wareable.com/features/what-is-apple-researchkit-iphone-watch-everything-you-need-to-know-931
[20] *Microsoft*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.microsoft.com/en-us/industry/health?rtc=1
[21] *Nuance*. Accessed: Apr. 28, 2020. [Online]. Available: https://www.nuance.com/healthcare/ambient-clinical-intelligence.html#in-action
[22] *Azureapifhir*. Accessed: Apr. 28, 2020. [Online]. Available: https://azure.microsoft.com/pl-pl/services/azure-api-for-fhir/
[23] *Microsoft Iomt Fhir*. Accessed: May 28, 2020. [Online]. Available: https://github.com/Microsoft/iomt-fhir
[24] *Microsoft Fhir Server*. Accessed: May 28, 2020. [Online]. Available: https://github.com/microsoft/fhir-server
[25] *1Up Health Platform*. Accessed: May 28, 2020. [Online]. Available: https://1up.health/
[26] *Onfhir.io*. Accessed: May 28, 2020. [Online]. Available: https://onfhir.io/
[27] *Bridgera Monitoring*. Accessed: May 28, 2020. [Online]. Available: https://bridgera.com/iot-digital-healthcare-software-solutions/
[28] *Docboxmed*. Accessed: May 28, 2020. [Online]. Available: https://docboxmed.com/
[29] M. Wozniak and D. Polap, "Intelligent home systems for ubiquitous user support by using neural networks and rule-based approach," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2651–2658, Apr. 2020.
[30] A. Sobecki, J. Szymański, D. Gil, and H. Mora, "Deep learning in the fog," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 8, 2019, Art. no. 1550147719867072, doi: 10.1177/1550147719867072.
[31] *Hl7fhir*. Accessed: Apr. 28, 2020. [Online]. Available: https://hl7.org/fhir/
[32] G. B. Moody, W. K. Muldrow, and R. G. Mark, "A noise stress test for arrhythmia detectors," *Comput. Cardiol.*, vol. 11, no. 3, pp. 381–384, 1984.
[33] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
[34] J. Achten and A. E. Jeukendrup, "Heart rate monitoring," *Sports Med.*, vol. 33, no. 7, pp. 517–538, 2003.

ANDRZEJ SOBECKI received the Ph.D. degree. He is currently with the Department of Computer Architecture, Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology. His research project concerns collecting and processing data describes patients with heart diseases. Based on the textual description, he performs classification of the patient and prepares prediction. Besides the research in processing data, he involved with the project of creating intelligent queue of patients in order to minimize waste of resources. He also involved with the process of creating and deploying the national anti-plagiarism system. His research interests include block chain technology and its applications, especially in healthcare.

JULIAN SZYMAŃSKI received the Ph.D. degree. He is currently with the Department of Computer Architecture, Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology. He addresses the problems of knowledge representation, methods of lexical knowledge acquisition, and linguistic data utilization. His research results mainly find applications in web search engines and systems of automated text categorization. He manages research projects for processing the information in Wikipedia. His main goal is to extract and structuralize the textual knowledge and construct interfaces capable of communicating in natural language. Besides his research in information retrieval domains, he involved with projects related to cognitive science. His mainstream of this research is building semantic memory models that improve natural language processing. He also involved in the area of analysing human emotions, by mining EEG signals. His results are implemented in the form of brain-machine interfaces for disabled people. He also works on analysing data with the IoT domain. He implements a set of sensors for an acquisition of data from beehives that allows one to predict apiary development and detect abnormalities. He served for many international conferences, including the International Conference on Networked Digital Technologies (NDT), the International Symposium on Innovations in Intelligent Systems and Applications (INISTA), and the International Conference on Parallel and Distributed Systems (ICPDS). His research interest includes application of data-mining methods to natural language processing (NLP). He was a reviewer for international journals and conference proceedings.

DAVID GIL received the Ph.D. degree in computer science from the University of Alicante, Spain, in 2008. He is currently an Associate Professor with the Department of Computing Technology and Data Processing, University of Alicante. He has published papers in high-quality international conferences, such as IJCNN, SAC, HEALTHINF, DCAI, SCAI, SAIS, and so on. He has also published articles in highly cited international journals, such as *Expert Systems With Applications* and *Applied Soft Computing*. He was involved with the organization of several international workshops, such as MoDIC, in 2012, and MoBiD, from 2013 to 2014. His research interests include applications of artificial intelligence, data mining, data warehouses, multidimensional databases, OLAP, design with UML, and MDA. He was a program committee member of several conferences and workshops, such as DAWAK, ARES, and CAiSE. He is a reviewer of several journals, such as *Neurocomputing*, *Expert Systems*, and *Soft Computing*.

HIGINIO MORA (Member, IEEE) received the B.S. degree in computer science engineering, the B.S. degree in business studies, and the Ph.D. degree in computer science from the University of Alicante, Spain, in 1996, 1997, and 2003, respectively. He has been a Faculty Member with the Computer Technology and Computation Department, University of Alicante, since 2002, where he is currently an Associate Professor and a Researcher with the Specialized Processors Architecture Laboratory. He has participated in many conferences. His work has been published in international journals and conferences with more than 60 published papers. His research interests include computer modeling, computer architectures, high-performance computing, embedded systems, the Internet of Things, and cloud computing paradigm.