Classification of Episodes in the Breathing Signal Using Machine Learning

Marcin Auer, Michał Mierzwa, Małgorzata Janik, Paweł Janik

University of Silesia in Katowice, Faculty of Science and Technology, Sosnowiec, Poland
marcin.auer@us.edu.pl, michal.mierzwa@us.edu.pl, malgorzata.janik@us.edu.pl, pawel.janik@us.edu.pl

DOI
https://doi.org/10.18690/um.fov.3.2026.1

ISBN
978-961-299-124-1

Machine learning methods, which have become increasingly popular in recent years, can be successfully used in the analysis of biomedical signals. Open-source libraries enable the creation of artificial intelligence models using, among others, configurable neural networks. The publication presents an approach to the classification of episodes in the breathing signal using accelerometer and pulse oximeter modules. In particular, it examines the influence of the type of network activation function and time window parameters, i.e. width and offset, on the model sensitivity. With the most optimally selected parameters, it was possible to obtain sensitivity of 85.71% for the detection of episodes in the signal and 100% sensitivity for the classification of calm breathing. The paper also discusses the possibility of creating intelligent sensors by implementing minimized machine learning models on miniature mobile devices with limited hardware resources. Moreover, it proposes a further research path, which is the development of adaptive algorithms able to independently select optimal learning parameters.

Keywords:
machine learning, signal analysis, TensorFlow, signal classification, episodes detection, biomedical signals

University of Maribor Press

1 Introduction

Artificial intelligence has been an intensively explored phenomenon in recent years, which has resulted in rapid development in this field. One of the paths of artificial intelligence development are generative models, such as language models capable of creating texts or models that generate graphics. As they become increasingly advanced, their level of computational complexity increases and, consequently, the demand for computing power grows. The popularity of artificial intelligence has also allowed for blazing new trails for the most common discriminatory models.At the core of the set of computational methods using machine learning are, among others, classifiers and anomaly detectors. These methods are successfully used in biomedical engineering, both in the field of image analysis - detecting, for example, tumours in magnetic resonance imaging (Kian et al., 2025), management of large databases used in diagnostics (Kwaśniewska et al., 2017), and analysis of continuous signals constituting time series (Backhus et al., 2025), such as, detection of anomalies in the ECG (electrocardiogram) signal (Ruiz-Barroso et al., 2025). A critical issue, particularly prominent in the field of image analysis, is the proper preparation of a training dataset. This process entails a huge demand for memory, and even single poorly prepared sample and data of inappropriate quality affect the accuracy of the entire model (He & Jinzhuo, 2025). Thanks to the selection of appropriate machine learning methods and adequate preparation of a training dataset, it is possible to apply artificial intelligence methods in many areas that have not used these solutions so far.

One of the important areas of artificial intelligence application and verification of the effectiveness of selected methods is the development of open-source libraries, such as TensorFlow. The open-source code allows for further customization of the library code to precisely defined applications and provides great support to the community of people using these libraries in their studies (Alquadah & Moussavi, 2025). The TensorFlow library includes tools for converting created artificial intelligence models to the TensorFlow Lite format. The size of the converted model is significantly reduced; the length of variables is reduced to the minimum necessary to limit the consumption of program memory. TensorFlow Lite makes it possible to optimize computational operations for the most effective use of built-in processor instructions, reducing the number of its cycles necessary to complete the operation,

allowing for the application on single-threaded units with low core clock speeds (Ramadan et al., 2025; Khalid et al., 2026; Ficco et al., 2024).

Reducing the demand for hardware resources opens the way to implementation on miniature, energy-saving, mobile platforms equipped with a microcontroller. SoCs (System-on-a-chip), i.e. systems built in a reduced architecture and integrating all necessary computer components, significantly lower the entry threshold thanks to their wide availability and low cost. Due to their small dimensions and low demand for electric energy, they are used for creating mobile sensor systems (Janik et al., 2019; Wang et al., 2025).

The versatility of this type of devices allows for the implementation of appropriately optimized artificial intelligence models. This means creating the next generation of intelligent sensors and wearable devices operating in accordance with the concept of the Internet of Things (Wang et al., 2023; Janik et al., 2025; Noorali et al., 2025; De Giovanni et al., 2023; Osa-Sanchez et al., 2025), intended for monitoring body parameters, detecting irregularities and alerting about critical conditions.

2 Measurement system and software environment

A measurement system has been built for the acquisition of the signal used for training and testing machine learning models. The system includes a digital accelerometer ADXL345, which natively supports communication via the Inter-Integrated Circuit serial bus (I2C). The MAX30102 heart rate and blood oxygenation (SpO2) sensor module was used as the reference sensor, which also supports communication via the I2C bus. This sensor allows SpO2 data to be compared with data from the accelerometer. The accelerometer sampling rate was set at 25Hz. The values were read using the NodeMCU v3 programmable board, containing the ESP8266 system. Data acquisition was performed using a PC.

The measurement cycle was carried out with the participation of a volunteer. The accelerometer and SpO2 sensor were placed in miniature housings designed and manufactured using 3D printing technology. The sensors in the housings are shown in Figure 1. The accelerometer was placed at the height of the sternum, whereas the SpO2 sensor was placed on the index finger. Measurements from both sensors were recorded concurrently in a lying position.

The open-source TensorFlow library (Abadi et al., 2015) was used to create machine learning models. The native scripting language calling library functions is Python. Anaconda Navigator was used as the platform to run the script interpreter. This platform extends the core Python language to include tools used in the field of Data Science and data analysis.

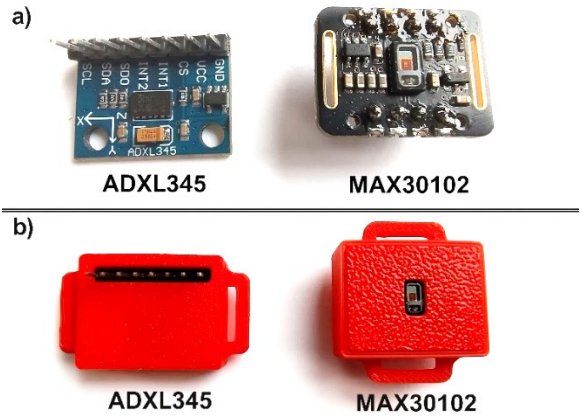

Figure 1: ADXL345 accelerometer and MAX30102 pulse oximeter a) unit modules, b) modules in 3D printed housings.
Source: Own

3 Training dataset

In order to prepare a training dataset, a cycle of calm breathing measurements was carried out, in which specific episodes were introduced: apnoea and faster breathing (approximately twice the frequency of breathing cycles compared to calm breathing). An eleven-minute waveform of the calm breathing signal was selected. It contained 3 apnoea episodes and 3 episodes of faster breathing of different durations. The episodes of faster breathing were as follows:

1 - six breathing cycles with a total duration of 36 s, 2 - two cycles with a total duration of 12 s and 3 - single respiratory cycle with a duration of 6 s.

The results of measurements recorded using an accelerometer are presented

in Figure 2, which shows the raw values for three axes (x, y, z).

Data from the accelerometer were processed with a resolution of 10 bits and a measurement range of ± 2G.

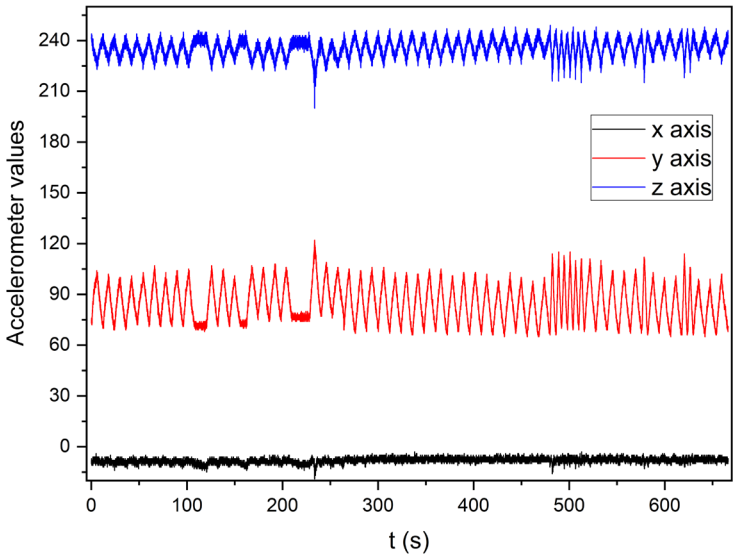

Figure 2: Training dataset - Raw accelerometer values
Source: Own

The waveforms of signal component values for reflected red (R) and infrared (IR) light returned by the SpO2 sensor were filtered. A low-pass filter with a cut-off frequency of 0.8Hz was used. The waveforms of the components coming from the pulse oximeter after filtration are illustrated in Figure 3. Using recorded fragments of signals from the accelerometer and filtered waveforms from the pulse oximeter, a file constituting a training dataset was created. The training data file, where the first line contains the headings of subsequent columns, had the following structure:

- Timestamp (time in ms),
- x, y, z (accelerometer components),
- R (the value of reflected red light returned by the pulse oximeter),
- R_LPF (R value filtered with a low-pass filter with a cut-off frequency of 0.8Hz),
- IR (the value of reflected infrared light returned by the pulse oximeter),

- IR_LPF (IR value filtered with a low-pass filter with a cut-off frequency of 0.8Hz),
- Label – class label to which correctly classified signal fragments are assigned, i.e.: Calm – calm breathing, Apnoea – breath holding, Faster – faster breathing,
- Split – a flag for the division between training and test data.

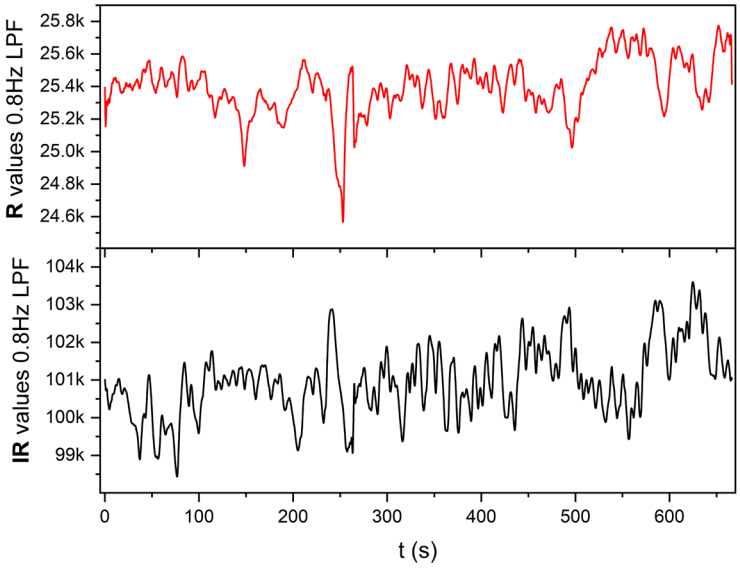

Figure 1: Training dataset - Filtered SpO2 sensor values

Source: Own

4 Machine learning parameters

The Keras sequential classifier model using an artificial neural network with the following structure was used for the TensorFlow model training process:

- Input layer – it accepts signal features sequentially and passes them to the next layer,
- LSTM (Long Short-Term Memory) bidirectional recurrent network layer with 128 units – it is characterized by high efficiency in training patterns taking into account the context,

- Dropout layer with a threshold value of 50% - it randomly drops half of the values stored in nodes to deepen the learning process,
- Dense layer composed of 128 neurons with the ReLU (Rectified Linear Unit) activation function described by (1) – it performs the final classification of the input data portions,
- Output layer with the selected activation function.

The network structure diagram is shown in Figure 4.

100 learning epochs were carried out for each of the training samples, and the selection of validation data was done randomly, selecting 10% of the input dataset.

$$ReLU(x) = \begin{cases} 0 \; for \; x < 0 \\ x \; for \; x \geq 0 \end{cases} \quad \text{.....(1)}$$

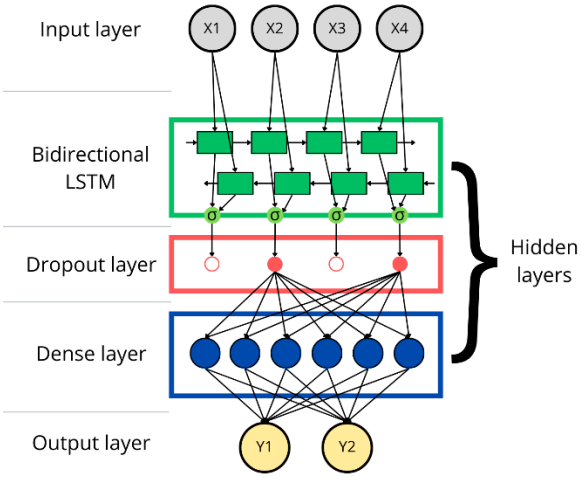

Figure 2: Neural network scheme
Source: Own

4.1 Division of the input data into windows

The time series constituting the input signal is divided into windows. This technique makes it possible to reduce the computational complexity as a smaller fragment of the recorded waveform is subject to classification (Alejandro, 2023). Another aspect

is to facilitate real-time signal analysis. A specific, fixed number of samples, which go to the input of the neural network with each iteration of the classifier, are loaded.

Data windowing introduces two additional parameters: window width and window offset. The data window concept is illustrated in Figure 5. The window width determines how many samples are included in one signal fragment. The offset determines the distance between the beginnings of subsequent windows. This means that in the case of implementing a classifier model or an anomaly detector on a single-threaded system intended to perform real-time classification, the window offset should be equal to its width.

A window offset value smaller than its width allows a certain part of the signal to be covered by several windows, which may contribute to higher model accuracy, especially at larger window width values. However, it requires concurrent classification of the signal and acquisition of its next fragment.

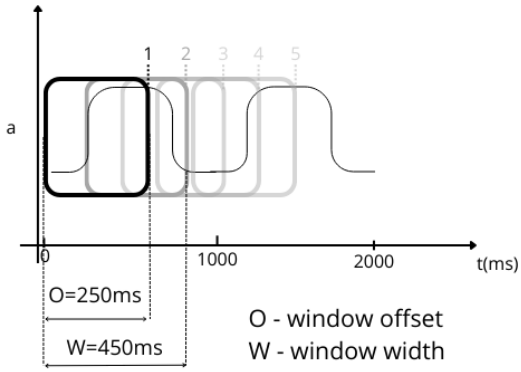

Figure 3: Illustration of window parameters
Source: Own

4.2 Selected window parameters and output layer activation functions

The maximum window width was set at 250 samples, which is approximately a period of calm breathing. The window width was reduced with particular emphasis on intermediate values: 150 samples – approximately the period of faster breathing, 75 samples - approximately half of the period of faster breathing. Table 1

summarizes the selected combinations of window parameters for which training tests of the neural network were carried out and its sensitivity was assessed.

Table 1: Selected window parameters combinations

Window offset (nubmer of	Window width (number of samples)										
	8	10	20	30	40	50	75	100	150	200	250
	4	5	10	10	10	10	15	25	25	25	50
	8	10	20	15	20	25	25	50	50	50	75
				20	30	50	50	75	75	75	100
				30	40		75	100	125	100	125
									150	125	150
										150	175
										175	200
										200	

Source: Own

The normalized softmax exponential function (2) and the sigmoid function (3) were selected as the activation functions of the output layer. A characteristic feature of the softmax function is the summation of all probabilities to the value 1, whereas when the sigmoid function is used, the probability is returned for each class in the range $(0,1)$.

$$softmax(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \quad \text{.....(2)}$$

where:

$\mathbf{z}$ – vector of real numbers, $\mathbf{z} = (z_1, \dots, z_K) \in R^K$,
K – number of elements of vector $\mathbf{z}$, $K \geq 1$,
$i = 1, \dots, K$.

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad \text{.....(3)}$$

5 Obtained results

Table 2 summarizes the sensitivity results (4) of the model when classifying calm breathing for the softmax and sigmoid activation functions. For the softmax calling function, the mean model sensitivity was 94.13 ± 5.79%. The lowest value was

71.43% for the window width of 200 and the offset of 175. The sigmoid function, in turn, provided a mean sensitivity of 93.97 ± 5.11% with the lowest score of 76.47% for the window width of 75 and the offset of 75.

$$sensitivity = \frac{TP}{TP+FN} \quad (4)$$

where:

TP – true positive result,
FN – false negative result.

Table 2: Model sensitivity in calm breathing classification in comparison with the window parameters. White rows represent the softmax function. Grey rows represent the sigmoid function.

Window offset	Window width										
	8	10	20	30	40	50	75	100	150	200	250
4	96.99%										
	97.89%										
5		98.11%									
		98.11%									
8	97.60%										
	98.90%										
10		96.99%	93.94%	96.18%	93.85%	92.25%					
		95.49%	92.42%	96.95%	95.38%	96.12%					
15				94.25%			97.65%				
				94.25%			97.65%				
20			87.88%	89.55%	92.31%						
			90.91%	91.04%	87.69%						
25						96.15%	84.62%	96.00%	100.00%	93.48%	
						92.31%	94.23%	100.00%	97.92%		
30				93.18%	90.91%						
				93.18%	90.91%						
40					90.63%						
					93.75%						
50						92.31%	92.59%	96.00%	100.00%	95.65%	95.65%
						88.46%	92.59%	92.00%	91.67%	95.65%	90.91%
75							82.35%	88.24%	93.75%	93.75%	100.00%
							76.47%	100.00%	93.75%	93.75%	93.33%
100								100.00%	91.67%	92.31%	81.82%
								100.00%	100.00%	84.62%	90.91%
125									100.00%	100.00%	90.00%
									100.00%	88.89%	100.00%
150									100.00%	100.00%	100.00%
									100.00%	87.50%	87.50%
175										71.43%	100.00%
										85.71%	100.00%
200										100.00%	100.00%
										100.00%	100.00%

Source: Own

Acceptable model sensitivity results (>50%) when classifying both types of episodes were obtained for window widths of [20, 200] and offsets of [10, 75]. Significant differences were observed between the tested activation functions, yielding a mean sensitivity of 53.96 ± 17.71% for the softmax function and 48.58 ± 15.22% for the sigmoid function within the range of parameters limited by the above intervals. The sensitivity values depending on the window parameters for these intervals are presented in Table 3.

Table 3: Model mean sensitivity in faster breathing and apnoea episodes classification in comparison with the window parameters

			Window width							
		Function	20	30	40	50	75	100	150	200
Window offset	10	softmax	55.36%	77.33%	62.50%	63.33%				
		sigmoid	55.36%	63.33%	59.38%	60.00%				
	20	softmax	41.96%	85.71%	68.75%					
		sigmoid	41.96%	71.43%	61.61%					
	25	softmax				66.67%	75.00%	33.33%	33.33%	50.00%
		sigmoid				66.67%	75.00%	58.33%	41.67%	
	30	softmax		40.00%	75.00%					
		sigmoid		50.00%	58.33%					
	40	softmax			50.00%					
		sigmoid			37.50%					
	50	softmax				50.00%	25.00%	33.33%	16.67%	66.67%
		sigmoid				50.00%	25.00%	33.33%	50.00%	33.33%
	75	softmax					75.00%	50.00%	50.00%	50.00%
		sigmoid					25.00%	25.00%	50.00%	25.00%

Source: Own

6 Conclusions

The conducted research showed the potential to use machine learning methods included in the TensorFlow library for detecting episodes in time series representing biomedical signals. However, the model sensitivity is highly dependent on the window parameters. Dividing the signal into time windows is necessary to enable classification of the signal waveform in real time. Configurability of model and learning process parameters, such as neural network topology, activation functions, learning force or time window parameters, allows the library to be used to analyse signals of various types. Converting models to TensorFlow Lite format is a direct path to implementation on devices with limited resources, such as microcontrollers.

Further optimization of the code in terms of resource use and efficiency will make it possible to obtain miniature, mobile, energy-saving intelligent sensors. An important issue is the correct selection of time window parameters for signal analysis. Their selection was carried out using an iterative method, checking the sensitivity of the model for each of the selected sets of window width and offset values. This process requires a lot of time and computing power, and, consequently, also electric energy.

The solution to the problem of variable parameters in hardware implementations are adaptive methods, which have been successfully developed in many areas (Li et al., 2025; Pielka et al., 2022). Developing an adaptive method so that the algorithm will be able to independently tune the learning parameters to the analysed signal can significantly shorten the learning process, and the resulting models will be more accurate.

References

Abadi M., Agarwal A., Barham P., Brevdo E., Chen Z., Citro C., Corrado G. S., Davis A., Dean J., Devin M., Ghemawat S., Goodfellow I., Harp A., Irving G., Isard M., Jia Y., Jozefowicz R., Kaiser L., Kudlur M., … Zheng X. (2015). TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems., https://tensorflow.org, doi: 10.5281/zenodo.4724125

Alejandro Z. Z. (2023). Data windowing, a technique used in time series forecasting using machine learning. https://www.linkedin.com/pulse/data-windowing-technique-used-time-series-forecasting-alejandro [access: 29.01.2026]

Alquadah A. M., Moussavi Z. (2025). A Review of Deep Learning for Biomedical Signals: Current Applications, Advancements, Future Prospects, Interpretation, and Challenges, Computers, Materials & Continua 2025;83(3), doi: 10.32604/cmc.2025.063643

Backhus J., Rao A. R., Venkatraman C., Gupta C. (2025). Time Series Anomaly Detection Using Signal Processing and Deep Learning, Applied Sciences, 15, 6254, doi: 10.3390/app15116254

De Giovanni E., Forooghifar F., Surrel G., Teijeiro T., Peon M., Aminifar A., Alonso D. A. (2023). Intelligent Edge Biomedical Sensors in the Internet of Things (IoT) Era, Emerging Computing: From Devices to Systems, Computer Architecture and Design Methodologies, p. 408-433, doi: 978-981-16-7487-7_13

Ficco M., Guerriero A., Milite E., Palmieri F., Pietrantuono R., Russo S. (2024). Federated learning for IoT devices: Enhancing TinyML with on-board training, Information Fusion, vol. 104, 102189, doi: 10.1016/j.inffus.2023.102189

He S., Jinzhuo W. (2025). Computational biomedical imaging: AI innovations and pitfalls, Medicine Plus, vol. 2, iss. 2, doi: 10.1016/j.medp.2025.100081

Janik M. A., Pielka M., Kovalchuk P., Mierzwa M., Janik P. (2025). Three-Dimensional Printed Stimulating Hybrid Smart Bandage, Sensors, 25, 5090, doi: 10.3390/s25165090

Janik P., Pielka M., Janik M. A., Wróbel Z. (2019). Respiratory monitoring system using Bluetooth Low Energy, Sensors and Actuators A Physical, 286:152-162, doi: 10.1016/j.sna.2018.12.040

Khalid S., Ur Rehman M., Usman A. B., Khawar M. (2026). Chapter 5 - Resource-efficient models for edge devices, Edge Intelligence – Advanced Deep Transfer Learning for IoT Security, p. 111-153, doi: 10.1016/B978-0-44-338297-0.00011-8

Kian A. Huang B. S., Neelesh Prakash M. D. (2025). Identifying Tumors on MRI Using TensorFlow: A Proof of Concept Machine-Learning Model, Neuromodulation: Technology at the Neural Interface, vol. 28, iss. 7, Supplement, page 13, doi: 10.1016/j.neurom.2025.08.026

Kwaśniewska A., Giczewska A., Rumiński J. (2017). Duże zbiory danych w zdalnej diagnostyce medycznej z wykorzystaniem technik głębokiego uczenia, VIII Krajowa Konferencja Naukowa: Infobazy, pages 1-8

Li Y., Zhang W., Zhang Z., Shi X., Li Z., Zhang M., Chi W. (2025). An adaptive compensation strategy for sensors based on the degree of degradation, Biomimetic Intelligence and Robotics, vol. 5, iss. 4, doi: 10.1016/j.birob.2025.100235

Noorali S., Baldwin J. C., Vaseashta A. (2025). Next-Generation Sensors Using Artificial Intelligence as Enabling Technologies for Biomedical and Healthcare Applications, Proceedings of the Eight International Symposium on Dielectric Materials and Applications (ISyDMA'8), doi: 10.1007/978-3-031-84809-4_17

Osa-Sanchez A., Ramos-Martinez-de-Soria J., Mendez-Zorrilla A. (2025). Wearable Sensors and Artificial Intelligence for Sleep Apnea Detection: A Systematic Review, J Med Syst 49, 66, doi: 10.1007/s10916-025-02199-8

Pielka M., Janik P., Janik M. A., Wróbel Z. (2022). Adaptive Data Transmission Algorithm for the System of Inertial Sensors for Hand Movement Acquisition, Senors 22(24):9866, doi: 10.3390/s22249866

Ramadan M. N. A., Ali M. A. H., Khoo S. Y., Alkhedher M. (2025), Federated learning and TinyML on IoT edge devices: Challenges, advances, and future directions, ICT Express, vol. 11, iss. 4, p. 754-768, doi: 10.1016/j.icte.2025.06.008

Ruiz-Barroso P., Castro F. M., Miranda J., Constantinescu D. A., Atienza D., Guil N. (2025). FADE: Forecasting for Anomaly Detection on ECG, doi: 10.48550/arXiv.2502.07389

Wang C., He T., Zhou H., Zhang Z., Lee C. (2023). Artificial intelligence enhanced sensors - enabling technologies to next-generation healthcare and biomedical platform, Bioelectric Medicine, vol. 9 art. 17, doi: 10.1186/s42234-023-00118-1

Wang S., Dong M., He J., Wu G., Li X., Pan X., Wu J., Bao R., Pan C. (2025). A multi-user wearable waistband system for real-time health monitoring of respiration, ECG, and body temperature, Microelectronic Engineering, vol. 299, 112346, doi: 10.1016/j.mee.2025.112346