LPRLC: Linear Predictive Run Length Coding to Improve Energy Consumption of WBANs

Mahdieh Hajiloo Vakil
Shahid Rajaee Teacher Training University
Zahra Shirmohammadi
shirmohammadi@sru.ac.ir
Shahid Rajaee Teacher Training University

Research Article

Keywords: Wireless Body Area Network, data compression, energy conservation, LPC, RLE

Posted Date: May 13th, 2024

DOI: https://doi.org/10.21203/rs.3.rs-4371727/v1

Additional Declarations: No competing interests reported.

LPRLC: Linear Predictive Run Length Coding to Improve Energy Consumption of WBANs

Mahdieh Hajiloo Vakil, Zahra Shirmohammadi*

Master Student, Computer Eng. Department, Shahid Rajaee Teacher Training University, Tehran, Iran, hajiloo@sru.ac.ir

*Assistant Professor, Computer Eng. Department, Shahid Rajaee Teacher Training University, Tehran, Iran. shirmohammadi@sru.ac.ir

Abstract

The Wireless Body Area Networks (WBANs) are biosensors placed on the body, inside the body, and around it. Body Area Network (BAN) are designed in micro sizes and have limited resources. Biosensors sometimes have errors in data recording and faced with duplicate and noisy data in real time. Data redundancy causes a significant energy consumption in sending and receiving data in the sensor. One of the most effective ways to reduce data volume is to compress it to save more energy. To solve these problems, the Linear Predictive Run Length Coding method (LPRLC) is presented which is a combination of Linear Predictive Coding (LPC) for data prediction and Run Length Encoding (RLE) for data compression. The signals attained from biosensors include blood pressure systolic (BPsys), blood pressure diastolic (BPdias), Respiration, Oxygen, and Heart Rate, which are recorded as a time series. First, the received signal is predicted continuously, and then the error resulting from the actual signal and the expected signal is calculated. In the last step, the resulting error is compressed by the RLE algorithm and sent to the destination. To compare the criteria of Energy Conservation (EC) and Compression Rate (CR), Huffman, Arithmetic, and Lempel-Ziv-Welch (LZW) algorithms are placed instead of RLE. The results show that the RLE algorithm has an average of 98% energy saving and up to 70 times reduction of data volume compared to other algorithms, which has improved 6% in energy consumption and 9 times reduction of data volume.

Keywords: Wireless Body Area Network, data compression, energy conservation, LPC, RLE.

Introduction

With the increase in population and healthcare needs, the use of wireless body area network technology is enhanced [1]. WBAN can remotely monitor human health and hygiene. It records vital signs such as heart rate, blood oxygen, body temperature, etc. If conditions are abnormal, it sends an alarm message to the health service center and the doctor takes the necessary actions [2].

Nowadays, Information and Communication Technology (ICT) produces about 2.5% of global greenhouse gas publishing, which has challenged the achievement of the goal of zero emissions by 2050. The enormous increase in electronic devices, computers, server connections, and running applications by exchanging massive amounts of data in real-time has greatly contributed to the rise in energy consumption. The most energy-consuming of sensors occurs for processing and sending data. When the sensor battery drains, the sensor stops working and disrupts the network performance. Sensors that are placed inside the body require surgery to replace them when they run out of energy, and this is very expensive. Consequently, energy consumption management in WBANs is a very important challenge that should be considered [3].

One of the most important ways to improve energy consumption in WBANs is to reduce the volume of data sent using data compression [4]. Data compression in wireless networks divided into three classifications: 1) lossless 2) lossy 3) hybrid. Lossless algorithms are grouped based on entropy and dictionary. In lossless algorithms, the original data is not lost after compression and is restored as initial data. The lossy algorithms are divided into four groups: 1) predictive coding, 2) transformation coding, 3) compressed sensing, and 4) lightweight coding. In this category, the abundance of data is significantly reduced after compression, and the compressed data is not equal to the original data after recovery. Third the hybrid, which is divided from the combination of both lossless and lossy, or both from the same category [5][6].

In the data received from the wireless body network, there are many noisy, erroneous, and repetitive data, and the sensors do not measure the data accurately in some situations. To solve this problems, lossy and lossless compression are needed [5][6]. In [29], the presented hybrid method suggests prediction using LCF and compression error by Huffman. This method is implemented in different layers of the WBAN so that the prediction and the coding of the error happen in the sensor layer, it causes a significant reduction in the amount of data. The data is collected in cluster head or sink and data labeling happens after recovery using a backpropagation neural network. In H-RLEAHE [32], there is a combination of RLE and Huffman algorithms in different scenario, both of which are entropy-based. The RLE algorithm alone has a higher compression rate and outperforms than H-RLEAHE. In DDCA-WSN [20], using data collection, compression occurs in intermediate nodes through four algorithms: BWT [25], MTF [25], RLE [11][27], and Arithmetic. As a result, energy storage is specific to the intermediate nodes layer. Due to the use of the data collection method along with the activation of four compression algorithms, the computational complexity of this method is high. In general, the focus of this method is on the significant reduction of data volume.

In this regard to solve the problem of unimportant and repetitive data and high energy consumption in the sensors layer, a hybrid method is presented [31] [32]. LPRLE has three phases: prediction, error calculation, and error compression. In the first phase, linear prediction coding [31] is used according to the signals received from the biosensors that are continuously recorded for one minute. Assuming that each sensor has a predictor, the predicted signal is obtained by the predictor. In the second phase, the difference of the original signal stored in the memory is obtained from the predicted signal, and in the third phase, it is coded by the RLE algorithm and sent to the destination.

LPRLE is compared with different algorithms of lossless coding: Huffman [8], LZW [17], and Arithmetic [13]. Results show the LPRLE with LPC and RLE has the greatest energy saving on average of 98% by keeping the proper compression ratio in five biosensors than DDCA [20], SZ with Huffman [18] and H-RLEAHE [32].

The key contributions of this paper are:

1. LPRLC comprises lossy and lossless techniques adjusted to biosensor datasets, using the LPC to predict at the sensor layer and attain influential accuracy.

2. In the compression phase, because of the data duplication in sensors the RLE is used to reduce data redundancy.
3. LPRLC implemented on the sensor layer has little computational complexity and is appropriate for limited sensor resources.
4. Taking into account the energy consumption, prediction, and compression that occur in the sensor layer LPRLC has saved up to 98% of the energy by maintaining a appropriate compression rate.

The rest of the paper is formed as follows, First in the related work architecture of WBAN, previous research related to compression in WBAN is analyzed. The phases of LPRLC are described in detail in the proposed method. In the next section, the results of the LPRLC are compared with other methods. Finally discussions, and conclusions are advanced.

Related Works

The architecture of WBAN consists of four layers. The first layer includes sensors and coordinators. In this layer, vital signs from the body are received by sensors and adjusted by coordinators. The second layer inclusive of personal information acquisition devices linked to sensors by wireless communication. The third layer is where data is stored, processed, and managed. The data analyzed in the third layer is sent to the fourth layer through routers in the Internet space. In the fourth layer, medical data centers, emergency service centers, and doctor's desks are located for the final decision of the patient's condition [1][2][3]. The WBAN architecture is shown in Figure 1.

Sensor nodes in a wireless network consume energy for sensing, processing, and communication purposes [4]. Normally, the energy consumption for data transmission is significantly higher than the energy for sensing and processing it. The communication level in the sensor node consumes the most energy in the sensor [5]. While the wireless sensor network is built based on battery energy, the battery source has a strong relationship with the lifetime of the network [6][7]. As a result, it is necessary to manage the energy dissipation of each sensor to increase the lifetime of the network. As the amount of data transfer decreases, energy consumption also decreases [8].

Compression algorithms in body wireless network are divided into 3 main groups [5]: The first group is lossless algorithms which are based on entropy or dictionary. Among the famous entropy-based algorithms, Huffman, RLE and Arithmetic are. LZW algorithm is based on dictionary. Huffman coding by building a binary tree leads to a longer compression time than RLE algorithm [7].

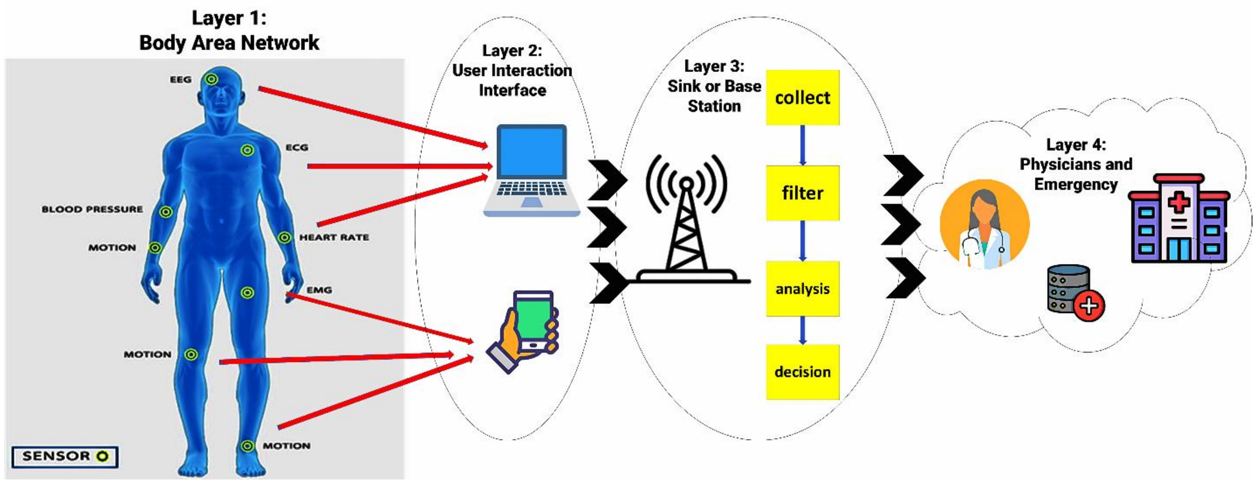

Fig 1: Architecture of WBAN

The time complexity of Huffman is equal to O (n log n), which is computationally more than RLE. The time complexity of RLE algorithm is equal to O (n). The main problem of RLE algorithm is that the compression results are influenced by the data source. RLE relies on data dependencies to achieve good compression rates. Huffman guarantees a suitable compression rate even when the data are not interdependent and unknown [7].

The second group is lossy algorithms, which are divided into four categories. Predictive coding that LPC and BPC are the most famous algorithms in this category [5]. Transform coding, DCT and DWT are based on this category. From lightweight coding, LTC can be mentioned and the last category is Compressed Sensing. The Discrete Wavelet Transform (DWT) and The Discrete Cosine Transform (DCT) are type of transformation coding. Energy consumption in DWT is high because of use of two low-pass and high-pass filters. DCT transforms the data from its initial domain to the alternating cosine domain. The resulting signal is a set of size and frequency. By comparing the DCT and DWT methods based on transformation, the compression rate in DCT and the performance of DWT in data recovery after compression is better. The number of missing packets and delays in the DWT is insignificant. In Bayesian Predictor Coding (BPC) [7] the data are predicted using a Bayesian predictor and the differences between the predicted data and the original data are coded. The efficiency of the prediction techniques is based on the accuracy of the predictor, and the smaller error gets the better compression rate. The Bayesian predictive coder is robust against data loss and fault. Linear Predictive Coding (LPC) [7][31] is a simple type of predictive method. This method uses a linear predictor to predict the data. LPC is suitable for data with linear characteristics. The compression rate of this method is almost the same as the BPC. The Compressed Sensing (CS) [10][11] method is suitable for sparse environments and compatible with low sampling rates. The data recorded in the wireless sensor network are in real-time, so by using a series of blocking methods like Block Sparse Bayesian Learning (BSBL), they can be adapted to the CS method. This type of data compression happens in the sensor node, so it should be proportional to the limitation of sensor resources in the network [10][11].

The third group of compression is a hybrid that is made from the combination of lossless- lossless [32][20], lossy- lossy, and from both groups lossless and lossy [21][25][29][34][50]. Today, many groups of hybrid methods are proposed to benefit from the advantages of both methods in reducing data volume. The H-RLEAHE [32] algorithm is one of the lossless-lossless hybrid methods. It has two phases, the first phase of data encoding through RLE and the second phase of tree construction using Huffman in the sender and receiver, both of those use the same initial node and show the nodes that are Not Yet Sent (NYT) with zero weight. Both encoding and decoding happen simultaneously. In this method, the scenario changes by relocation in two phases. In general, in terms of energy saving and compression rate, the performance of the RLE algorithm is better than Huffman, H-RLEAHE, and H-AHERLE. Sunyaev-Zel'dovich (SZ) [14][18] is a hybrid method that incorporate three phases: 1- data flattening. 2- data prediction by curve fitting approaches. 3- Using a lossless algorithm for coding the error. In [18], the curve fitting method, which includes 3 approaches: 1- Quadratic Curve Fitting (QCF) 2- Preceding Neighbor Fitting (PNF), and 3- Linear Curve Fitting (LCF) is used. After this step, by specifying a threshold, the approach that has the least error is used as a predictor, and in the last step, the errors are coded by Huffman. This method reduces energy consumption in the sensor. DeepSZ [21] compresses the Deep Neural Network (DNN) [24][21] by SZ. The point of this method is to distinguish the best error bound to attain a suitable compression ratio.

Proposed Method

The LPRLC method has three phases: 1- linear prediction by LPC, 2- calculation of predicted data error from original data, and 3- error coding using the RLE algorithm. The phases of the LPRLC method are shown in Figure 2.

Linear Predictive Run Length Coding (LPRLC)

Shouldering that each communication node has a predictor associated with its transmitter that operates on previously transmitted signals stored locally to produce an estimate of the future signal [31]. The linear predictive coding method is a continuous signal processing method X that is sampled at discrete time points i as a time series X(i). The basic idea of linear predictor coding is that each sample of the time series can be estimated as a linear combination of the previous samples as in Equation 1.

$$X(i) \approx R(i) = Y_1 X(i-1) + Y_2 X(i-2) + \cdots Y_n X(i-n) \qquad (1)$$

And the other word.

$$X(i) \approx R(i) = \sum_{j=1}^{n} Y_j X(i-j)$$

where R(i) is equal to the estimated signal of X(i) and n is the number of samples to estimate the signal. Y_j is equal to the approximate predictor coefficients for estimating the signal X(i). LPC is used in signal interpolation, signal recovery and noise reduction. The main signal is characterized by relatively small values of coefficients. LPC can be used for signal compression, where only the coefficients and the first n samples need to be stored or transmitted, and then the remaining signal is obtained from these values using (1) [31].

Finally, coefficients are a compressed representation of the original signal. Similar signals should have the same coefficients that can be used to identify signs or content by comparing them with known signals. As shown in Figure 1, first the main received signal is entered into the memory of the sensor, and the next signal is predicted by the predictor. In the next phase, the error resulting from the prediction signal is calculated from the original signal. Finally, the resulting error is coded by the RLE in Algorithm 1 and sent to the sink.

The RLE algorithm compresses data by removing redundancy. The way RLE works is that it first counts the number of repetitions of a character and then codes it as a number of repetitions with a character. For example, the input string produces "XXXXX" as a pair of 5X. In RLE the higher number of character repetitions, the higher compression rate. [9]. RLE flowchart is shown in Figure3.

Compressing the error significantly reduces the data volume compared to compressing the original signal, and this is one of the advantages of data prediction methods before sending it to the destination. On the receiving side, the error is recovered by the RLE and predicted by the predictor. The original signal is obtained from the sum of the error and the predicted. Finally, the original signal store in the receiver's memory in the same way to predict the next entry. To evaluate the presented method, in the last step, instead of the RLE algorithm, Huffman, Arithmetic and LZW algorithms have been placed. that the RLE algorithm has the highest energy storage and compression rate.

Experiment Results and Discussion

Python programming is used to run LPRLC on five biosensors including systolic blood pressure, diastolic blood pressure, respiration, oxygen, and heart rate, and Numpy, Matplot, Pandas, Librosa, and Sklearn libraries are called. Also, the fitter package in Python is used to find the appropriate probability distribution with the data set obtained from biosensors. The Fitter class is a main class of the fitter package that uses the get_distribution function to test different probability distribution methods on the data set and provides the best fit.

According to the results, sum square error criterion, which is obtained from the set of errors of the prediction,as shown in Figures 4-8 the probability distribution of the BPsys is gamma, the BPdias is Nakagami, Hart Rate is Skewcauchy, Oxygen is Hyperbolic and Respiration is Gen norm. All The probability distribution functions are the category of continuous probability distribution and related to the normal distribution. Linear predictive in this type of data set have good accuracy in prediction therefore the lower error rate is proportional to the compression rate increases[7]. As shown in Figure 9-13, the signal of biosensors was recorded as a continuous time series for one minute.

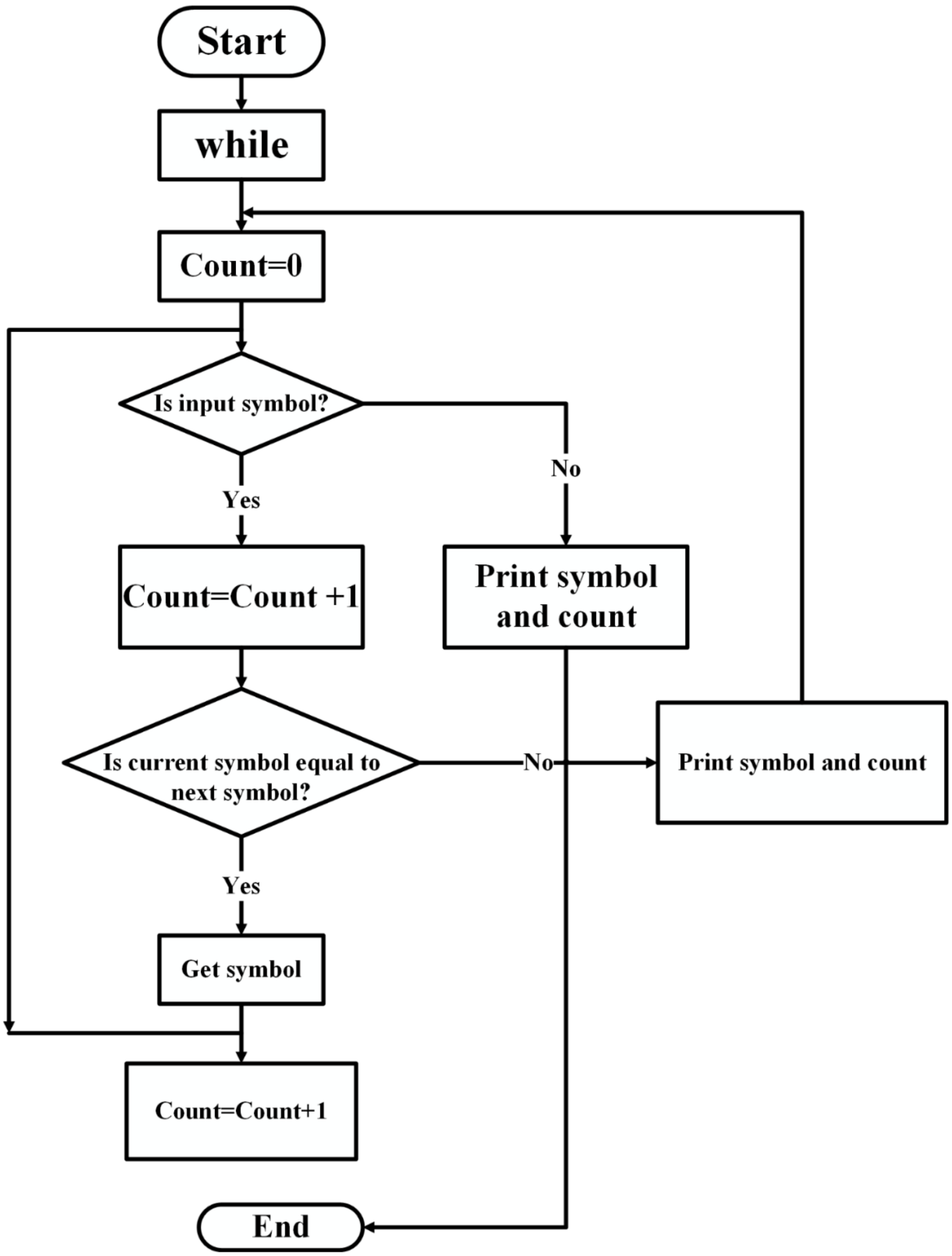

Fig 2: Flowchart of RLE

The first criterion measured according to (2). RMSE is used to assess the efficiency of the model and accuracy of predictor.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{original} - x_{predicted})^2} \qquad (2)$$

Where n is the number of datapoints.

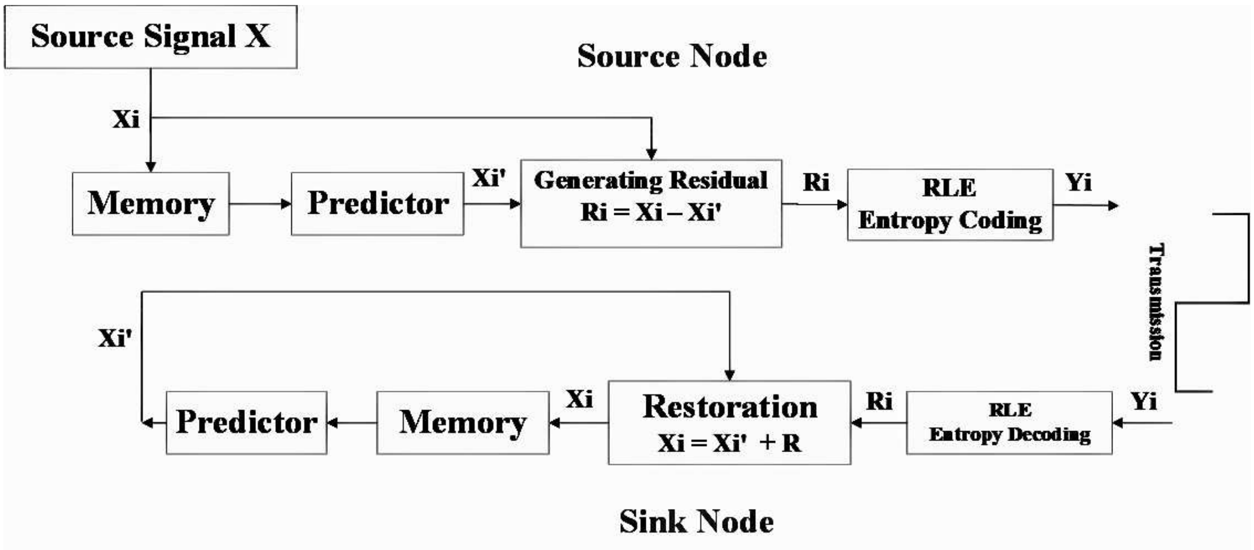

Fig 3. Structure of LPRLC

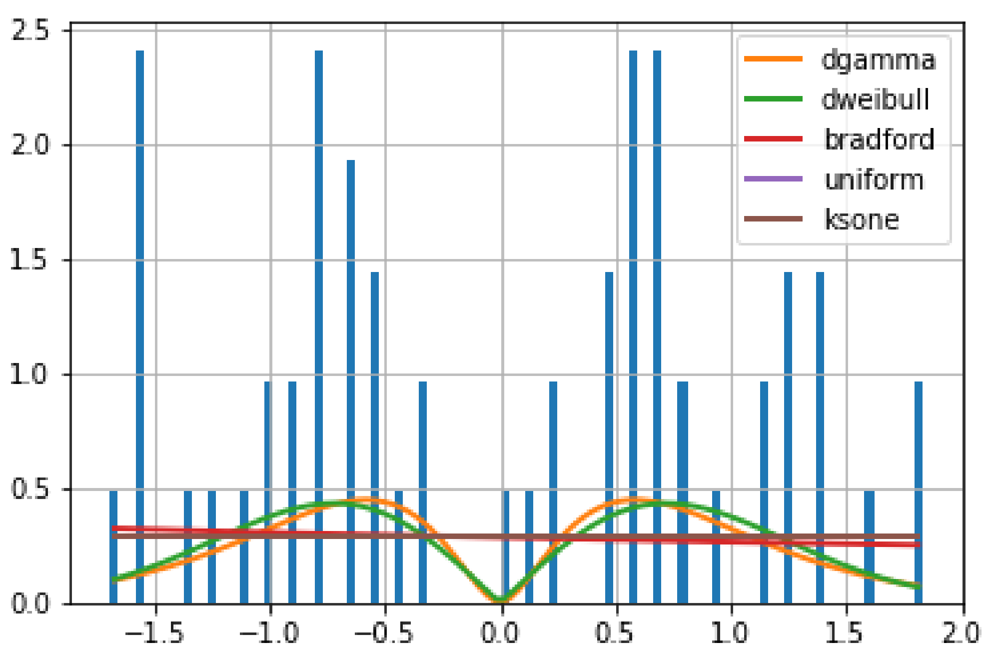

Fig 4: Probability distribution of BPsys is gamma.

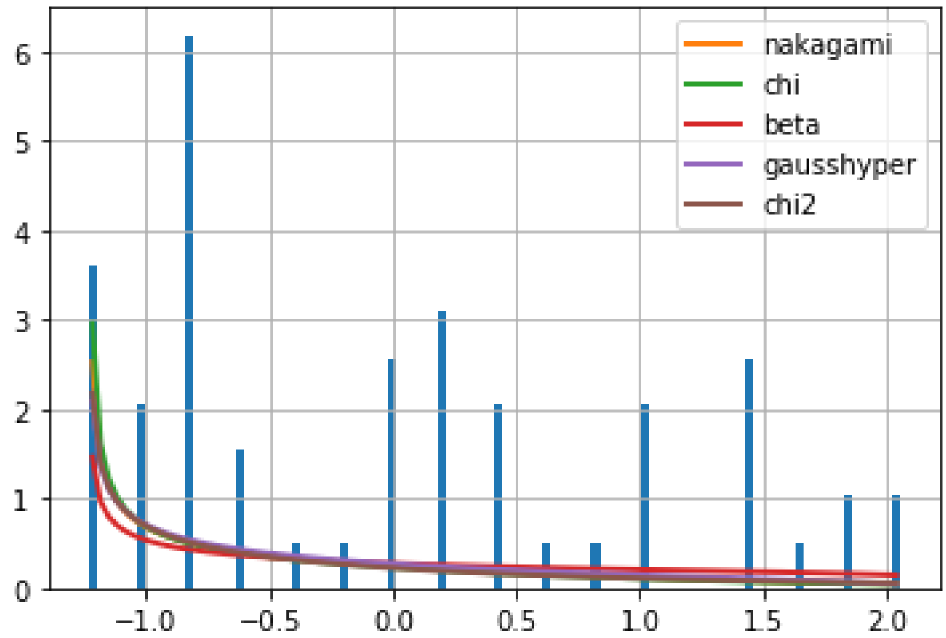

Fig 5: Probability distribution of BPdias is Nakagami.

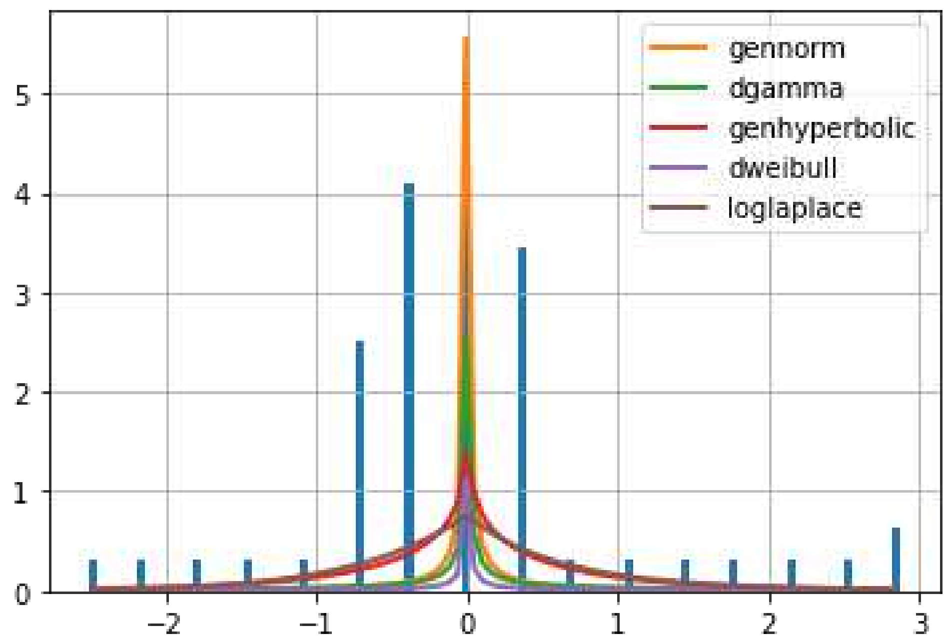

Fig 6: Probability distribution of Respiration is Gen norm.

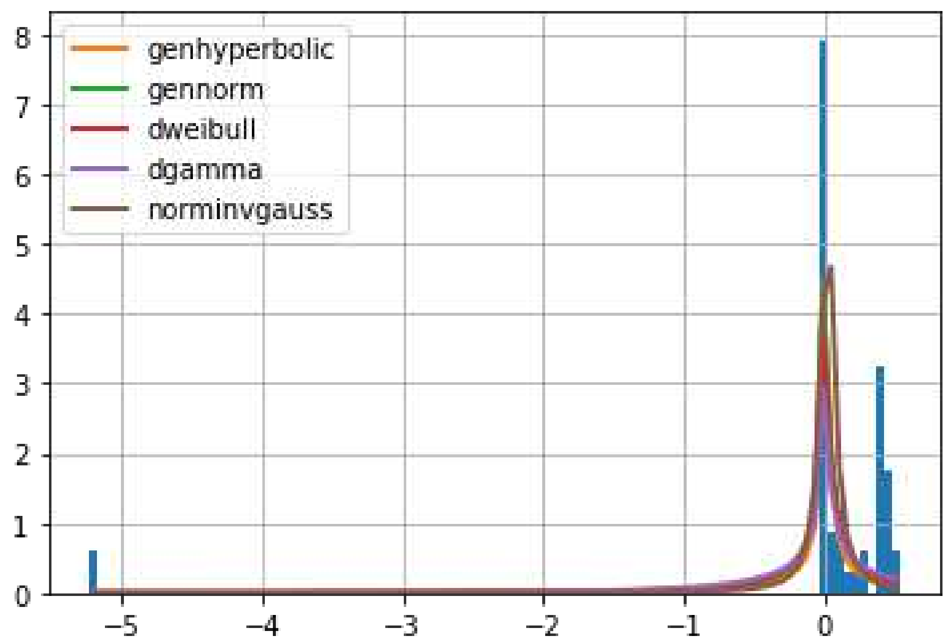

Fig 7: Probability distribution of Oxygen is Hyperbolic.

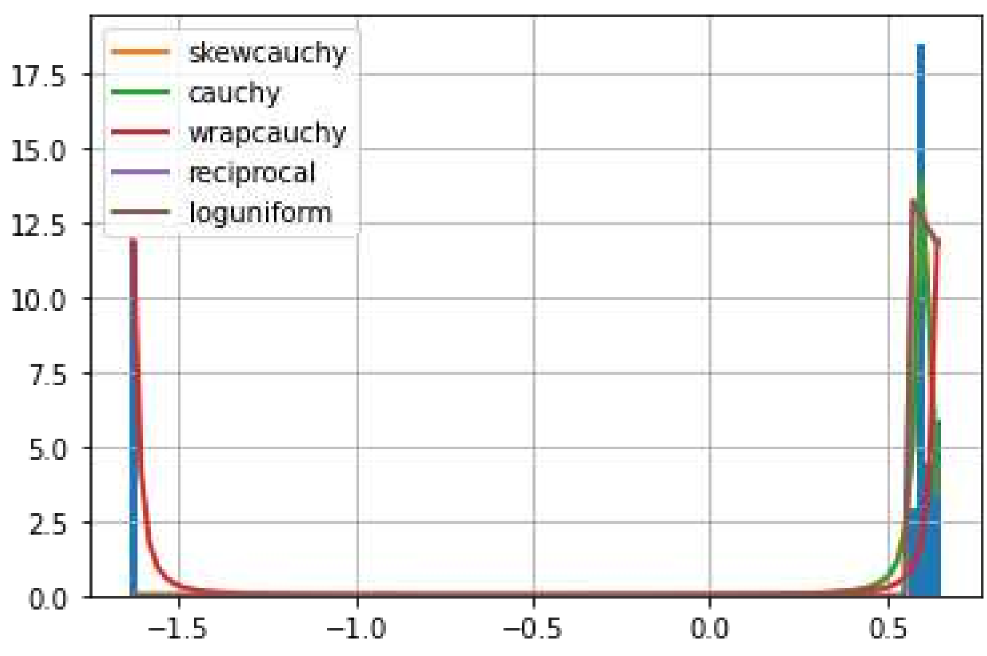

Fig 8: Probability distribution of Heart Rate is Skewcauchy.

In Figures 9,10,11,12 and 13, the signals of five biosensors with their prediction by LPC are shown. The larger the RMSE value, the greater the difference between the predicted and observed values. This means that a regression model fits the data worse. As reported in Table 1, the highest RMSE value is related to the oxygen sensor equal to 22.86 and the lowest value is 3.757 related to the breathing sensor.

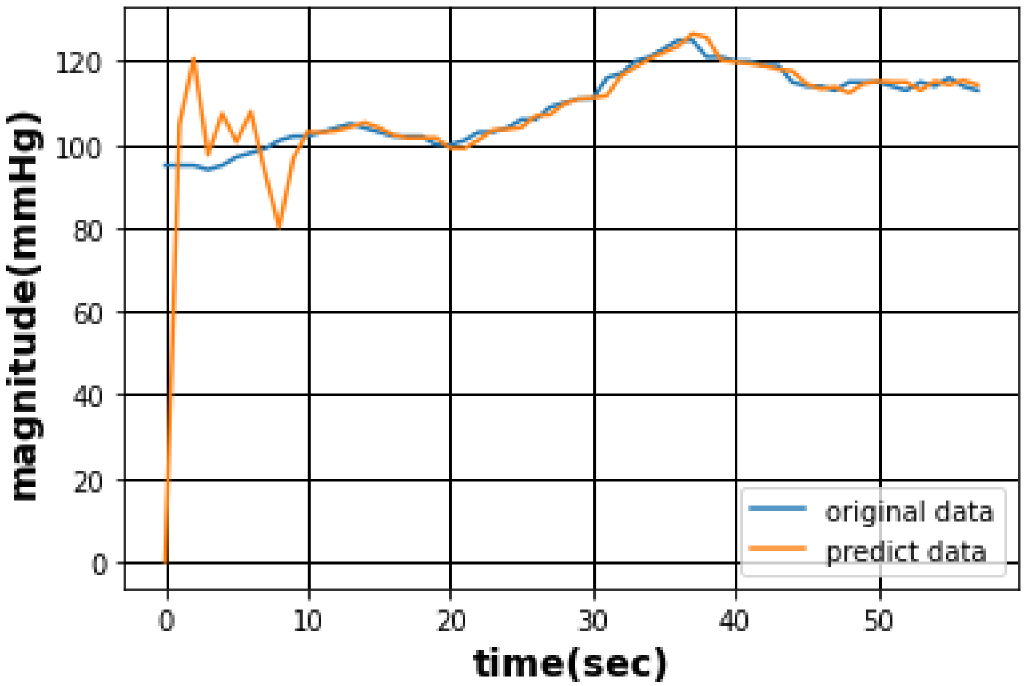

Fig 9: RMSE of LPC on BPsys

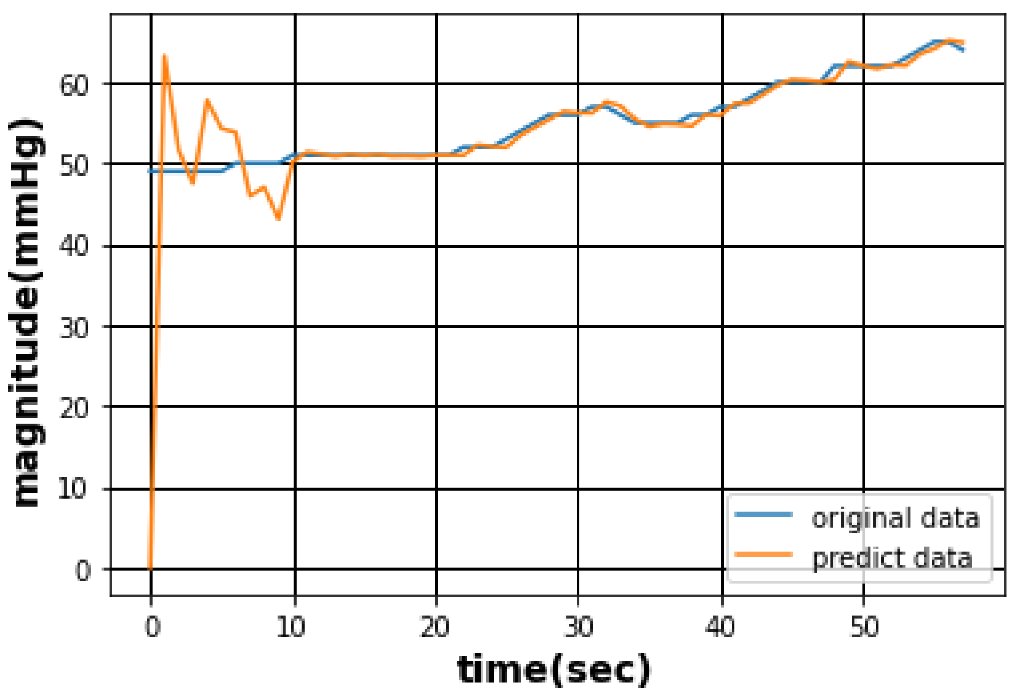

Fig 10: RMSE of LPC on BPdias

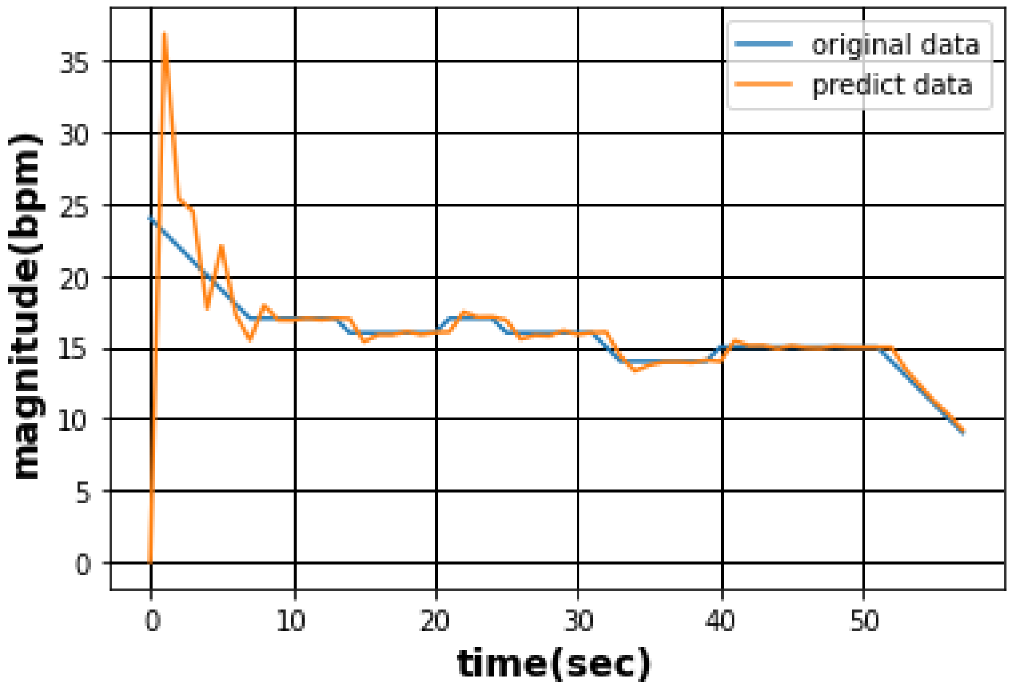

Fig 11: RMSE of LPC on Respiration

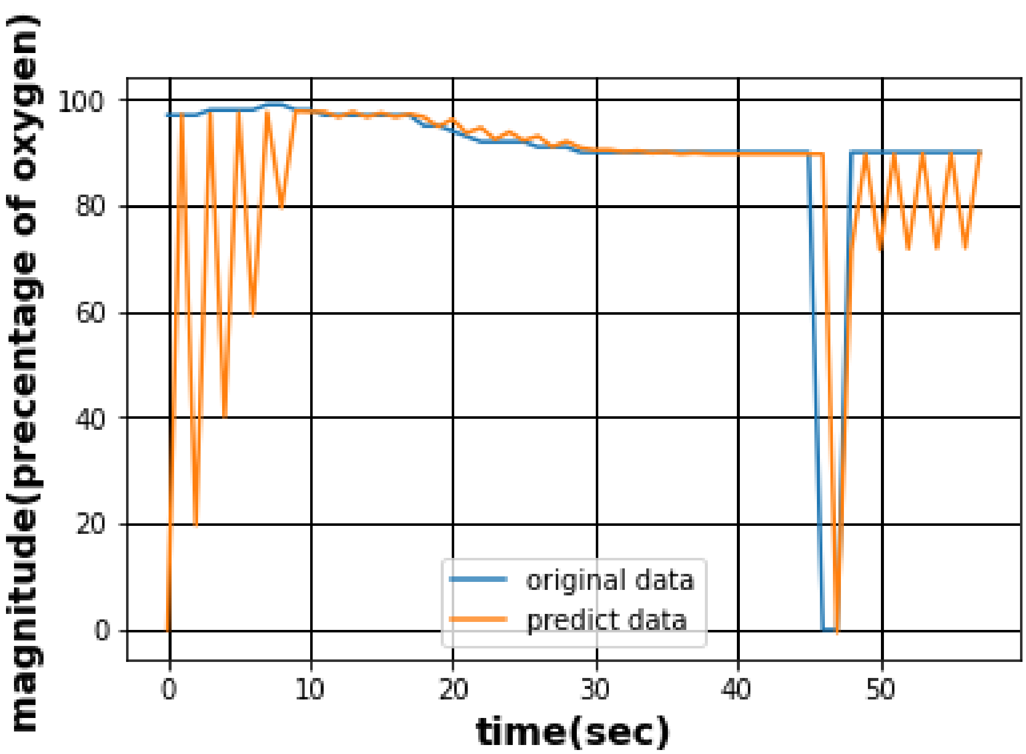

Fig 12: RMSE of LPC on Oxygen

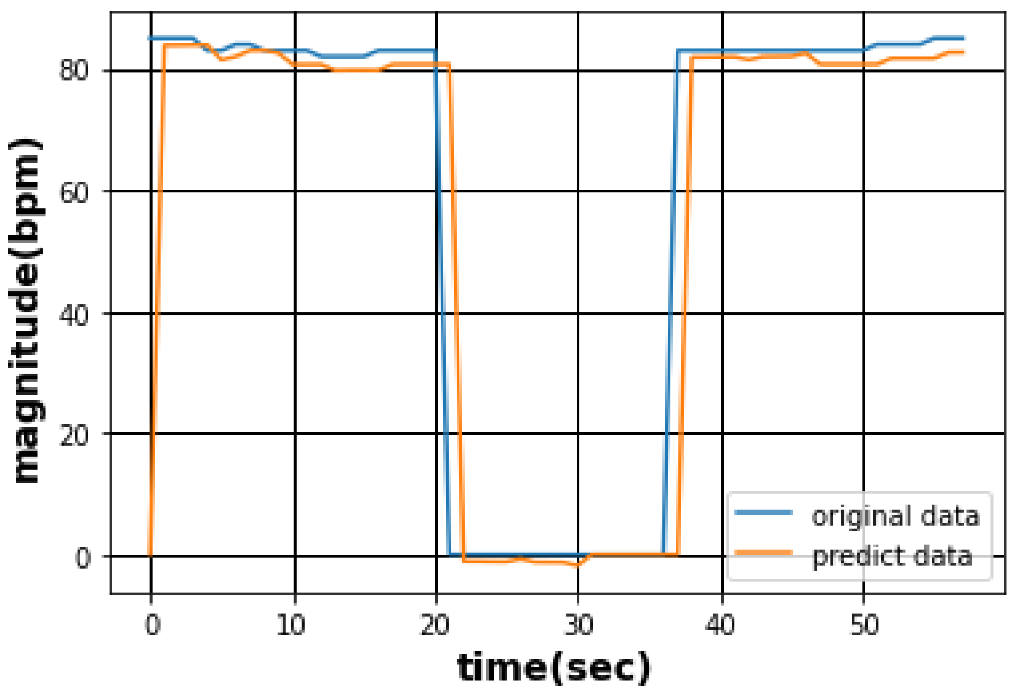

Fig 13: RMSE of LPC on Heart Rate

Table 1: RMSE of biosensors by LPC

Sensor	Score
Blood Pressure Systolic	13.55
Blood Pressure Diastolic	6.97
Respiration	3.75
Oxygen	22.86
Heart Rate	18.93

The second measured criterion of EC is calculated by equation 3.

$$\mathrm{EC} = \left(1 - \frac{1}{\mathrm{CR}}\right) \times 100 \qquad (3)$$

CR is compression ratio as defined in equation 4:

$$\mathrm{CR} = \frac{\text{Count of Uncompressed Data}}{\text{Count of Compressed Data}} \qquad (4)$$

The highest stored energy for RLE algorithm is equal to 98% on average and the lowest is equal to 76% on average for LZW algorithm. In detail, the amount of saved energy after compression in

Table 2 is shown. The comparison of stored energy values is shown in graphs 9,10,11,12and 13 for each biosensor.

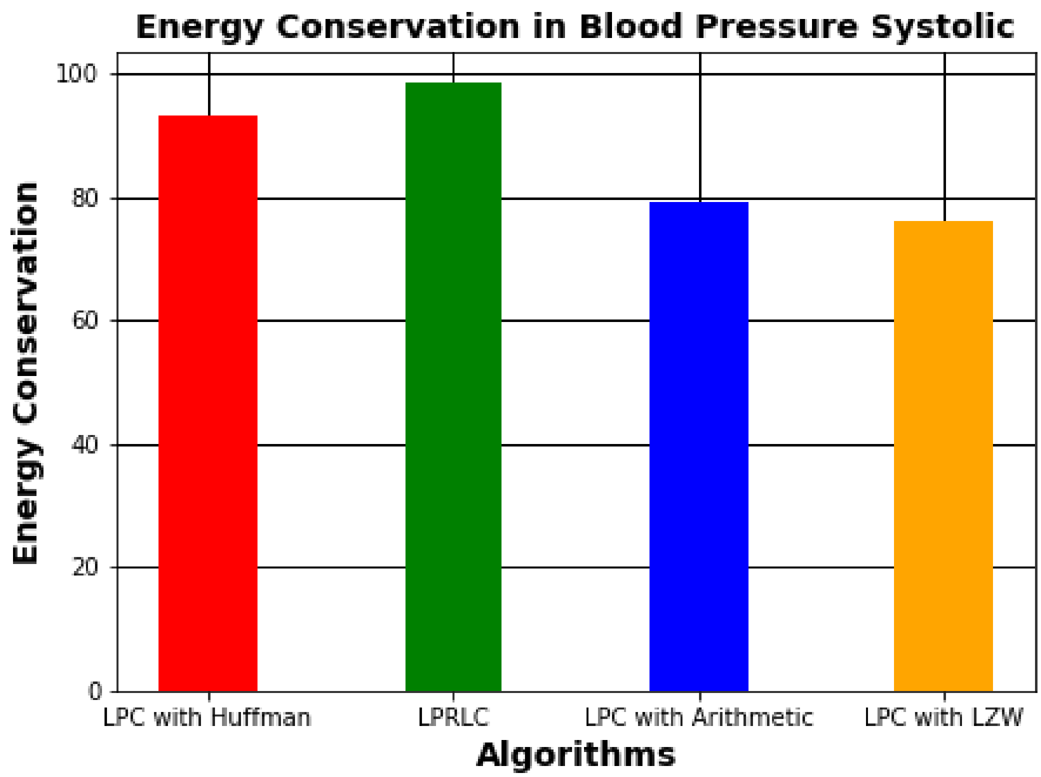

Fig 9: EC of each algorithm on BPsys

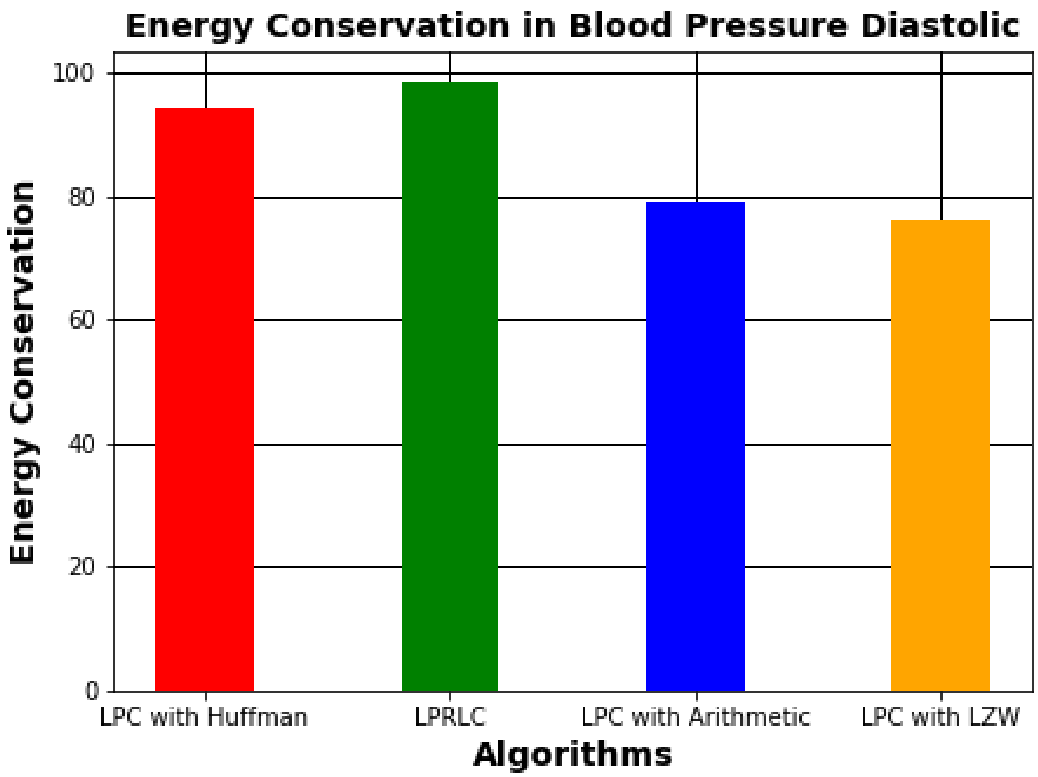

Fig 10: EC of each algorithm on BPdias

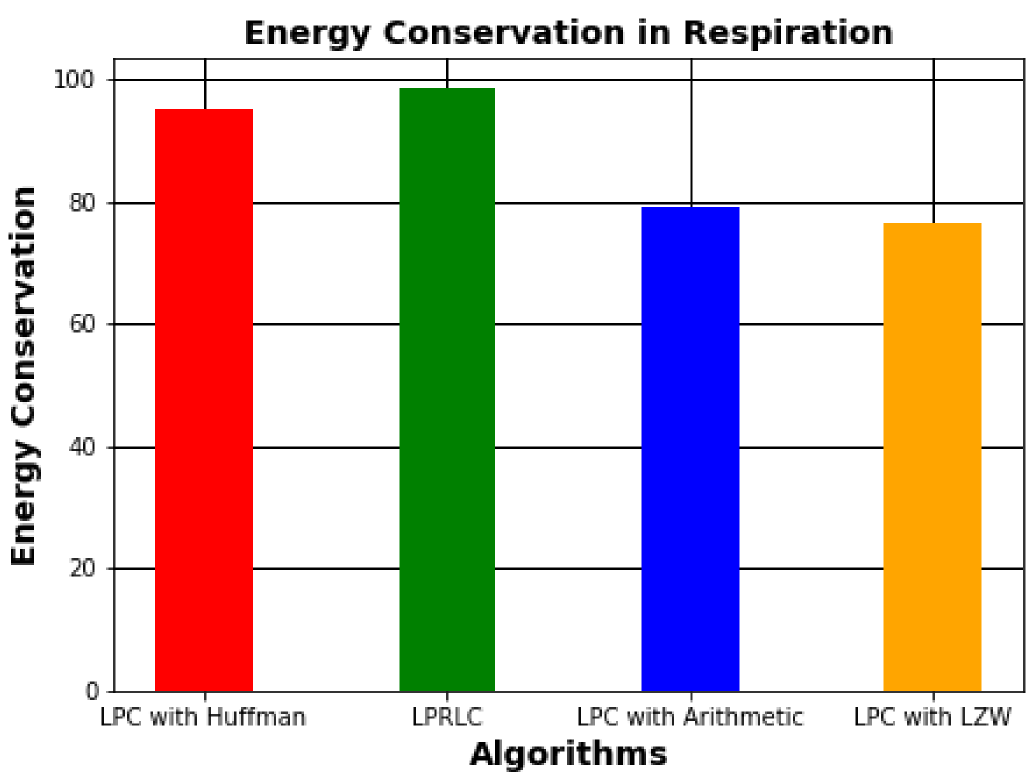

Fig 11: EC of each algorithm on Respiration

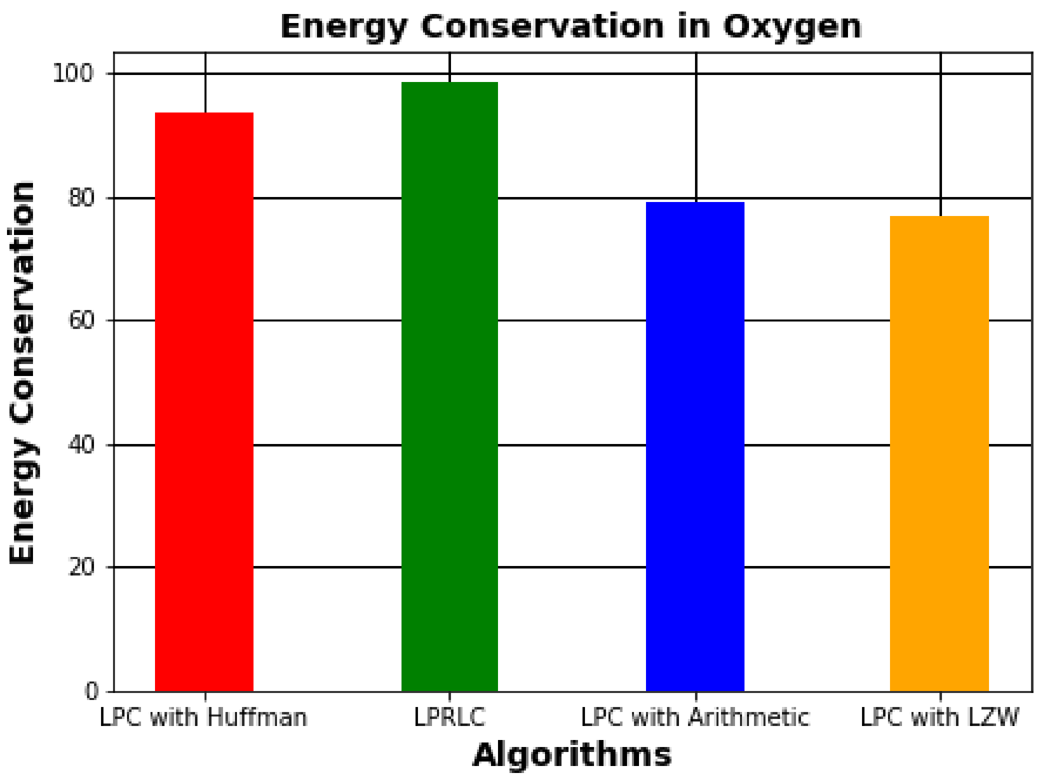

Fig 12: EC of each algorithm on Oxygen

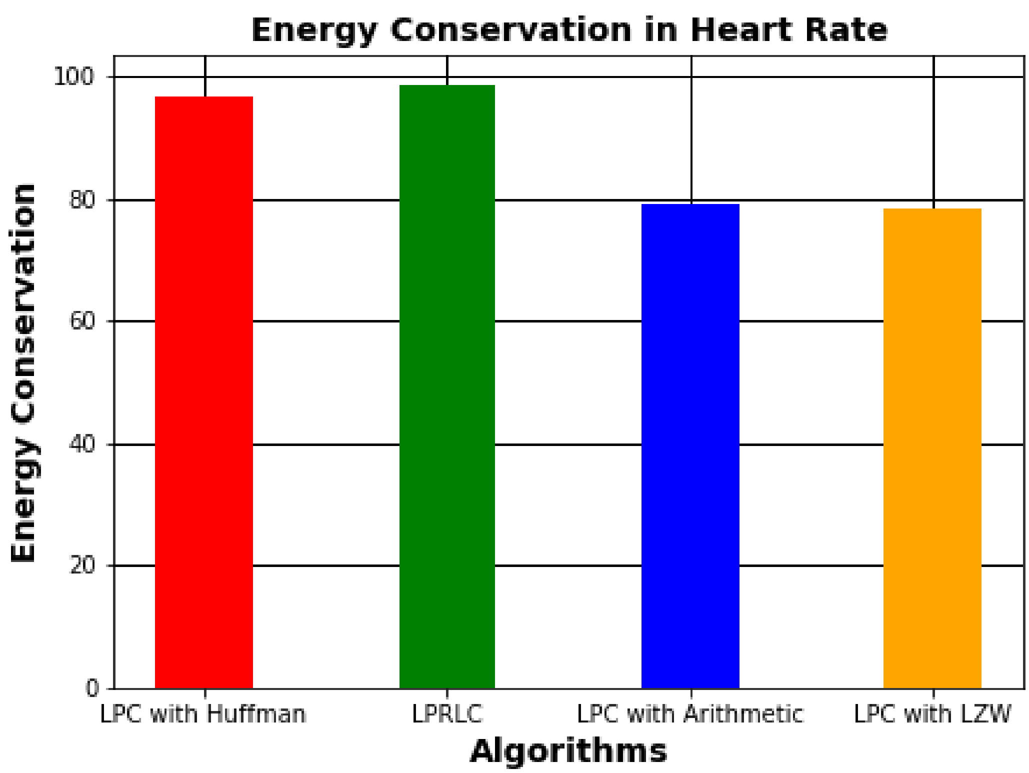

Fig 13: EC of each algorithm on Heart Rate

Table 2: EC of each algorithm on biosensors

Sensor	EC of RLE	EC of Huffman	EC of Arithmetic	EC of LZW
Blood Pressure Systolic	98.57%	93.04%	78.98%	76.27%
Blood Pressure Diastolic	98.54%	94.28%	79.09%	76.1%
Respiration	98.57%	95.17%	78.98%	76.48%
Oxygen	98.55%	93.58%	78.93%	76.64%
Heart Rate	98.60%	96.82%	78.98%	78.36%

The compression rate criterion calculated using Equation 4 is shown in Table 3. The highest compression value equal to 70.42 belongs to the RLE algorithm, in other words, the LPRLC method. The lowest compression value on average equal to 4.32 belongs to LZW algorithm. According to the obtained results and existing relationships, the compression rate is proportional to the amount of energy storage.

LZW is a dictionary-based reversible algorithm. This algorithm requires a lot of memory and is not suitable for sensor limitations. With the increase of inputs for processing, its speed decreases. The Arithmetic and LZW have the worst results in compression rate and energy saving. According to the high number of operations and calculations in Arithmetic, the execution speed is slow. Arithmetic needs the whole code word to decode each input, this may cause it to error to retrieve the code. The compression rate results show that the LZW algorithm as a dictionary-based method in the third phase is not compatible with the LPC method. Considering the structure of LPC, it is better to place an entropy-based algorithm in the compression phase. For this reason, LZW achieves the lowest amount of stored energy and the lowest compression rate.

According to the reports, the best results are obtained for RLE and Huffman algorithms. The RLE requires less memory than Huffman to store information, and the execution time of the RLE is faster than Huffman. RLE relies on inter-data dependency, and the more repeatable and interdependent data will achieve the best performance in compression. Basically, in RLE, compression results are influenced by data sources. The data set from biosensors in this research has high repetition and reliance. Thus, a very good performance of the RLE algorithm is obtained .

Table 3: CR of each algorithm on biosensors

Sensor	CR of RLE	CR of Huffman	CR of Arithmetic	CR of LZW
Blood Pressure Systolic	70.42	14.49	4.76	4.21
Blood Pressure Diastolic	68.96	17.54	4.78	4.34
Respiration	70.42	20.83	4.76	4.25
Oxygen	71.42	15.62	4.76	4.29
Heart Rate	70.92	32.25	4.76	4.62

Conclusion

Nowadays, the use of body wireless network technology has increased significantly. Sensors in the WBAN have a limited source of energy. The most energy consumption in sensors is related to data exchange. One of the most effective solutions to reduce data volume and save energy is data compression. In this article, the combined LPRLC method is presented. This method uses a combination of linear predictive coding for prediction and runtime coding for compression. For comparison in energy storage and compression rate, lossless algorithms such as Huffman, Arithmetic, and LZW are placed. RLE algorithm performs best compared to other algorithms in energy saving and compression rate. On average, the RLE 98%, and Huffman 95% have the greatest amount of energy saving, and the Arithmetic algorithm 78% and LZW 76% have the lowest amount of energy saving. The highest compression rate is related to the RLE algorithm and the lowest is related to the LZW algorithm. According to the coding structure of the linear predictor, the coding part by the entropy-based algorithm has a better performance in data compression, so the LZW algorithm has not shown a good performance in data compression.

References

[1] K.G. Mkongwa, Q. Liu, Ch. Zhang, F.A. Siddiqui, "Reliability and Quality of Service Issues in Wireless Body Area Networks: A Survey," *Signal Processing Systems*, vol. 7, no. 1, pp. 26-31, March 2019.

[2] K. Hasan, K. Biswas, N.S. Nafi, Md.S. Islam "A comprehensive review of wireless body area network," *Network and Computer Applications*, vol. 143, pp.178-198, 1October 2019.

[3] H. Sharma, A. Haque, F. Blaabjerg, "Machine Learning Wireless Sensor Networks for Smart Cities: A Survey," *Electronics*, vol.10, no. 9, pp.1-22, 23April 2021.

[4] R. Ali Khan, A.S.K Pathan, "The state-of-the-art wireless body area sensor networks:A survey," *Distributed Sensor Networks*, vol. 14, Iss. 4, pp. 1-23, April 2018.

[5] T. Sheltami, M. Musaddiq, E. Shakshuki, "Data compression in Wireless Sensor Networks,"*Future Generation Computer Systems*, vol. 64, pp. 151-162, 24January 2016.

[6] X. Lai, Q. Liu, X. Wei, W. Wang, G. Zhou, G. Han, "A Survey of Body Sensor Networks," *Sensors*,vol.13, no. 5, pp. 5406-5447, 24 April 2013.

[7] Ch. Chen, L. Zhang, R. L. K. Tiong, "A new lossy compression algorithm for wsn using Bayesian predictive coding,"*Wireless Networks*, vol. 26, no. 64, pp. 5981-5995 ,10 July 2020.

[8] H. P. Medeiros, M. C. Maciel, R. D. Souza, M. E. Pellenz, “Lightweight Data Compression in Wireless Sensor Networks Using Huffman Coding,” *Distributed Sensor Networks*, vol. 2014, Iss. 1, pp. 1-11, 23 January 2014.

[9] A. Saidani. et.al. , "A Lossless Compression Approach Based on Delta Encoding and T-RLE in WSNs," *Wireless Communication and Mobile Computing*", vol. 2020, 21 Sept 2020.

[10] Zh. Zhang, S. Makeig,”Compressed Sensing of EEG for Wireless Telemonitoring with Low Energy Consumption and Inexpensive Hardware,” *IEEE TRANSACTION ON BIOMEDICAL ENGINEERING*, vol.60, Iss. 1, pp. 221-224, 2 Nov 2013.

[11] Sh. Sun, J. Xing, Zh. Zhou, W. Wang, J. Chen, “Comparative Study of Compressed Sensing for Heart Sound Acquisition in Wireless Body Sensor Networks,” *IEEE Access*, vol. 8, pp. 22483 – 22492, 20 January 2020.

[12] A. Nasif, Z. A. Othman, N. S. Sani, ” The Deep Learning Solutions on Lossless Compression Methods for Alleviating Data Load on IoT Nodes in Smart Cities,” *Sensors*, vol. 21, no. 12, 4223, 20June 2021.

[13] F. Marcelloni, M. Vecchio, "An Efficient Lossless Compression Algorithm for Tiny Nodes of Monitoring Wireless Sensor Networks," *The Computer Journal*, vol. 52, no. 8, pp. 969-987, November 2009.

[14] Sh. Di, F. Cappello, " Fast Error-bounded Lossy HPC Data Compression with SZ," *International Symposium on Parallel and Distributed Processing (IPDPS)*, 21July 2016.

[15] A. Moon, J. Kim, J. Zhang, S. W. Son, " Lossy Compression on IoT Big Data by Exploiting Spatiotemporal Correlation," *IEEE High Performance Extreme Computing Conference (HPEC)*, September 2017.

[16] S. Jin, Sh. Di, X. Liang, J. Tian, D. Tao, F. Cappello, " DeepSZ: A Novel Framework to Compress Deep Neural Networks by Using Error-Bounded Lossy Compression," *Proceedings of the 28th International Symposium on High-Performance Parallel and Distributed Computing*, pp. 159-170, 17 June 2019.

[17] H. M. Al-Kadhim, H. S. Al-Raweshidy, " Energy Efficient Data Compression in Cloud Based IoT," *IEEE SENSORS JOURNAL*, vol. 21, no. 10, 15 May 2021.

[18] J. Liu, et al. ," Improving Lossy Compression for SZ by Exploring the Best-Fit Lossless Compression Techniques*," IEEE International Conference on Big Data (Big Data),* 13 January 2022.

[19] G. Liu, et al. ," Novel Data Compression Algorithm for Transmission Line Condition Monitoring," *energies*, vol. 14, issue. 24, 8 December 2021.

[20] E. T. Fute, et al. ," DDCA-WSN: A Distributed Data Compression and Aggregation Approach for Low Resources Wireless Sensors Networks," *International Journal of Wireless Information Networks*, vol. 29, pp. 80-92, 14 January 2022.

[21] J. Azar, et al. ," An energy efficient IoT data compression approach for edge machine learning," *Future Generation Computer Systems*, vol. 96, pp. 168-175, 11 February 2019.

[22] S. Karupusamy, et al. ," Effective energy usage and data compression approach using data mining algorithms for IoT data," *Expert System,* 10 April 2022.

[23] N. Jacob, P. Somvanshi, R. Tornekar, " Comparative Analysis of Lossless Text Compression Techniques," International Journal of Computer Applications, vol. 56, no. 3, pp. 0975-8887, October 2012.

[24] A. B. Said, et al. ," Multimodal deep learning approach for joint EEG-EMG data compression and classification," *IEEE Wireless Communications and Networking Conference (WCNC)*,19 March 2017.

[25] E. T. Fute, et al. ," An Efficient Data Compression Approach based on Entropic Coding for Network Devices with Limited Resources," *European Journal of Electrical and Computer Engineering*, vol. 3, no. 5, October 2019.

[26] B. Chreim, J. Nassar, C. Habib, "RADAR - Regression Based Energy-Aware DAta Reduction in WSN: Application to Smart Grids," *Information Networking and Applications,* pp. 1-14, July 2021.

[27] N. A. Khairi, A. B. Jambek, " Run-Length Encoding (RLE) Data Compression Algorithm Performance Analysis on Climate Datasets for Internet of Things (IoT) Application," *International Journal of Nanoelectronics and Materials,* vol. 14, December 2021.

[28] A. Kadhum, M. Al-Qurabat, "A Lightweight Huffman-based Differential Encoding Lossless Compression Technique in IoT for Smart Agriculture," *International Journal of Computing and Digital Systems*, vol. 11, no. 1,January 2022.

[29] J. Chen, et al. , "Data compression algorithms for sensor networks with periodic transmission schemes," *MATEC Web of Conferences*, January 2022.

[30] S. Pandey, et al. ," EEDCS: Energy Efcient Data Collection Schemes for IoT Enabled Wireless Sensor Network," *Wireless Personal Communications*, vol. 129, pp. 1297-1313. 23 Febuary 2023.

[31] M. W. Spratling, " A Review of Predictive Coding Algorithms," Brain and Cognition, vol. 112, pp. 92-97 ,March 2017.

[32] M. Mishra, et al. ," Investigation of Energy Cost of Data Compression Algorithms in WSN for IoT Applications," *Sensors*, vol. 22, no. 19, 1-28, 10 October 2022

[33] N. Mirewa, A. M. Ramly, A. Amphawan, T. K. Neo, "Optimizing Medical IoT Disaster Management with Data Compression," *Journal of Informatics and Web Engineering*, vol. 3, no.1, February 2024.