

Utilizing Machine Learning for Sensor Fault Detection in Wireless Sensor Networks

Abubakar Abdulkarim, Israel Ehile Ehile, Refik Caglar Kizilirmak

Dept. of Electrical and Computer Engineering, Nazarbayev University, Astana, Kazakhstan

Abstract—This paper discusses the utilization of machine learning for sensor fault detection in Wireless Sensor Networks (WSNs). The WSN comprises many wireless devices with limited processing power, battery life, and memory capacity. Successfully detecting faulty sensors within a WSN can lead to increased efficiency in the fault detection system, reduced network traffic, and lower power consumption. To enhance the network management, researchers sought a technique for detecting sensor defects. In this paper, machine learning techniques are employed to address the issue of failure detection in WSNs. The utilization of machine learning techniques, specifically Kernel Support Vector Machine (SVM) and Artificial Neural Network (ANN), are demonstrated. The paper further compares the performances of the chosen machine learning algorithms in classifying sensor data as faulty or fault-free. The problem is treated as a binary classification problem. The findings of this study contribute to the development of effective fault detection systems in WSNs.

Index Terms—fault detection, WSN, SVM, ANN, machine learning

I. INTRODUCTION

The Wireless Sensor Network (WSN) system comprises several wireless devices with limited processing power, battery, and memory space. Each device in the network has sensing and limited processing capacities. This comprises sensors and actuators that monitor changes or occurrences in the environment and transmit information to the base station for analysis and subsequent action. The events include temperature, humidity, motion, air density, wind velocity, etc. WSNs have found extensive application in diverse sectors for control and monitoring purposes, such as the military sector, health sector, transportation, smart homes, etc. This includes traffic monitoring, environmental surveillance, patient monitoring, etc.

Sensor nodes are occasionally deployed in hazardous or unmonitored environments such as rain, forests, snow, volcanos, thunder, and wind [1]. Moreover, sensors can experience malfunctions and failures due to their electronic nature. These failures can be broadly classified into three types: communication failure, software failure, and hardware failure [2].

Faulty sensors in WSN may lead to a decrease in the efficiency of the fault detection mechanism, an increase in network traffic, and an increase in power consumption. Sensors devices have limited resources, and they are designed to operate for a long period, which can be from hours, days,

months, or years. Therefore, there is a need to conserve sensor power. In addition, it may not be practical to replace their batteries in case they are deployed in hazardous locations.

Since faulty sensors may increase network congestion by forwarding misleading and inaccurate data, a sensor fault detection mechanism is required for better management of the network. The detection mechanisms may enhance sensor data reliability and network bandwidth. However, a complex detection mechanism may increase the sensor's power consumption. Hence, there is a tradeoff between maintaining the service quality of the network and conservation of the sensor's energy [3]. Fault detection mechanisms detect and record faulty sensors in WSN; the record can be used to replace a faulty sensor, fault recovery process, or isolate the faulty sensor from the WSNs.

In this work, our aim is to utilize machine learning methods for fault detection of sensors in the WSNs. The problem is viewed as a binary classification problem; this is because the dataset used is labeled data of sensor readings that are either faulty or fault-free. The paper aims to assess the performance of different machine learning techniques in classifying faulty and fault-free sensor data.

II. PREVIOUS WORKS

Several research efforts have been conducted in the literature to deal with the issue of sensor fault detection in WSNs. The techniques proposed in the literature can either be a distributed approach, a centralized approach, or a combination of both which is known as a hybrid approach [4]–[6]. Some of those proposed methods are based on machine learning. An approach based on machine learning was introduced for the identification of sensor faults within WSN [7]. In this method, all sensors within the network transmit their data to a central node referred to as the sink node. For each sensor, the timestamps associated with its data are organized in descending order.

Furthermore, [8] proposed a fault detection mechanism in WSNs using an SVM classifier. The SVM classifier provides a decision function that separates faulty sensor readings from the correct ones. An experimental study is provided to show the performance of their machine learning-based approach. Detection accuracy and the false positive rate were considered as the accuracy metric to be used to evaluate the accuracy of the proposed machine learning-based detection method. The

accuracy of the proposed approach was studied using data collected from sensors deployed in a lab environment.

A distributed mechanism for isolating faulty sensors in WSNs has been proposed in [9]. This isolation mechanism allows for faulty sensors to function as routers only, but logically, they are disconnected from the network by the fault detection mechanism. In addition, the mechanism compares sensor data with their neighbors to improve the accuracy of the sensed data and isolate the faulty sensors from the network.

In contrast, a recurrent neural network-based distributed fault detection technique has been proposed in [10]. The technique requires collaborative work between the sensors in the network to detect faults. Authors in [11] focus on the classification and identification of system and data faults. A Markov model is utilized to capture the dynamics of the faulty data and the fault-free data. Moreover, a structural analysis of the Markov model is used to determine the kind of system fault or data fault that affects the measurements of the sensor devices.

Subsequently, a group of K-Nearest Neighbors is created based on the values of the sensor data. Euclidean distances are then computed for each data from the sensor, and any data points that deviate significantly from the norm are identified as potential faults. An artificial neural network-based method for detecting outliers is proposed in [12], [13]. The method uses temperature data captured by sensors in WSN. In the method, a temperature value is considered an outlier when there is a large difference between the predicted temperature value and the expected temperature value. This difference is considered an error value. The proposed method is an anomaly prediction with good performance in terms of prediction accuracy.

In [14], a hybrid prediction technique utilizing a Kalman filter and an Extreme learning machine is presented. The sink node was trained using faulty sensor data using the filter and the extreme learning machine, which served as the predictive classifier. The extreme learning machine has a low communication overhead and can make highly accurate predictions. Utilizing WSN data and artificial random anomalies introduced into the data, the performance of the suggested technique has been assessed. The performance metric used is the computational time and the detection accuracy.

In contrast, our paper utilizes the approach of two machine learning techniques to detect faults in WSNs, as opposed to other authors who used single machine learning approach in detecting faults. Our study assesses and compares the performances of two machine learning techniques in accurately classifying sensor data as either faulty or fault-free. This approach allows us to offer a comprehensive evaluation of the suitability and effectiveness of different machine learning approaches in the context of sensor fault detection, enabling researchers and practitioners to make informed choices based on their specific needs and constraints.

A. Sensor Faults

This section presents types of sensor faults. There are various types and causes of faults in WSNs. Sensor faults can be categorized broadly into two primary groups: faults based on location and those based on time. In time-based faults, the time of faults indicates the duration taken by the fault. Sometimes faults occur temporarily and it only takes a limited duration of time. Moreover, the time-based faults are further divided into transient faults and persistent faults. Transient faults do not stay permanently; they normally disappear quickly. Transient faults detection mechanisms have been proposed in [15]–[18]. On the other hand, persistent faults are the types of faults that occur permanently, they exist till a fault recovery is conducted. In most cases, persistent faults occur to a specific component in the network.

In general, the whole network is not faulty, faults affect only specific components in the network. Therefore, rather than the entire network, fault detection techniques must focus on certain network components. Data-centric faults and system-centric faults are two general categories of faults that may be distinguished based on where they occur. Assume that the data generated by a specific sensor can be described as $d(i_d, t, f(t))$, where i_d represents the sensor identity, t denotes the timestamp, and $f(t)$ is a function representing the sensor's value at the given timestamp t . Furthermore, $f(t)$ can be expressed as $\alpha + \beta x + \eta$, where α represents an additive offset, β signifies the multiplicative constant referred to as gain, x stands for the sensor's actual non-faulty value at the timestamp t , and η accounts for noise originating from external factors.

Several offset fault techniques have been discussed in the literature. An offset fault refers to a minor discrepancy in the original data, where it deviates by a specific constant additive value from the anticipated sensor readings. This type of fault can arise due to bad sensor calibration. The model of this type of fault can be expressed as $d' = \alpha + d + \eta$ where d is the original sensor data. Another type of sensor fault is called Gain Fault. Gain Fault occurs when there is a deviation in the rate of change of sensor data from the expected behavior over a certain period. In this type of fault, non-faulty sensor data get multiplied by some constant value to produce faulty sensor data. This fault is modeled as $x' = \beta x + \eta$ where x is the original non-faulty sensor measurement, while β is a constant multiplied by non-faulty sensor measurement due to reasons such as improper calibration of the sensor device. Other sensor faults include stuck-at, data loss, out-of-bounds and random faults [19].

III. METHODOLOGY

In this paper, we have compared the performance of two machine learning-based algorithms for sensor fault detection in WSN. The machine learning algorithms are SVM and ANN (a single-layered). SVM is originally designed for binary classification which is the problem targeted in this study. Binary classification deals with the classification between only two classes. Moreover, in this paper we have only two

classes of sensor data, the data is either faulty data or fault-free data. In SVM, the predictions only depend on a part of the training data which are close to the decision boundary. These subsets are known as support vectors, and they are the most important data points in SVM classification. Modifying the loss function and combining it with a kernel trick is termed a support vector machine. Support vector machine approaches binary classification problems using the concept of margin [20]. A margin in SVM is defined as the distance between the support vectors of the two classes that are separated by the decision boundary. Therefore, the aim is to choose a decision boundary that provides a maximum margin between the two classes. This is termed as the maximum margin classification.

For a two-class classification problem, the decision boundary is defined by the linear model, where $w^T \phi(x)$ represents a feature vector resulting from applying a feature transformation and b represents the bias term.

$$y(x) = w^T \phi(x) + b \quad (1)$$

where the distance from point x to the hyperplane defined by $y(x) = 0$ in a perpendicular direction is expressed as $|y(x)|/||w||$. For correct classification, the distance of a point x_n to the decision is determined as

$$\frac{t_n y(x_n)}{||w||} = \frac{t_n (w^T \phi(x_n) + b)}{||w||}. \quad (2)$$

The margin is determined by the perpendicular distance to the nearest point x_n within the dataset, and to maximize this distance we need to optimize the parameters w and b by solving the optimization equation given as

$$\underset{w, b}{\operatorname{argmax}} = \left\{ \frac{1}{||w||} \min_n [t_n (w^T \phi(x_n) + b)] \right\}. \quad (3)$$

However, solving the above equation directly is complex, an alternative problem that is more manageable should be derived. The kernel trick and the concept of Lagrange multipliers are employed to solve this problem.

To utilize the previously trained model for classifying new data points, the prediction is expressed in terms of the Lagrange multipliers instead of weights as

$$y(x) = \sum a_n k(x, x_n) + b \quad (4)$$

where a_n is the Lagrange multiplier and $k(x, x_n)$ is the kernel function. The algorithm for SVM shown in Algorithm 1 [21] iteratively trains a Support Vector Machine (SVM) by initializing support vectors, generating subsample spaces, evaluating Lagrange multipliers, determining optimal directions and biases, and ultimately obtaining the decision boundary.

A broad, useful approach to understanding discrete-valued, real-valued, or vector-valued functions from a collection of training samples is provided by artificial neural networks. ANN learning is an ideal option for issues where the training data correlates to noisy, complex sensor data because of its robustness to errors in the training data. Multilayer perceptron

employs algorithms such as the backpropagation algorithm. Gradient descent is a technique used by backpropagation algorithms to fine-tune network parameters to best suit a training set of input-output pairs [22].

The relevant cost function based on sum squared training errors is stated as follows for a network with many output units

$$E(\vec{w}) = \frac{1}{2} \sum_{d \in D} \sum_{k \in \text{outputs}} (t_{kd} - O_{kd})^2 \quad (5)$$

where O_{kd} is the network output linked to network output k and training example d , and t_{kd} is the target value. The weight values under consideration are determined using the gradient descent rule to minimize E . The backpropagation algorithm used to discover these weights is shown in Algorithm 2 [23]. In a typical application, the weight-update loop in backpropagation may be iterated thousands of times. To end the process, several termination criteria might be used.

Algorithm 1 SVM Algorithm [21]

- 1: Initialize support vectors and $k(x, x_n)$
 - 2: **for** $s \leftarrow 1$ to S **do**
 - 3: Generate subsample space
 - 4: Evaluate Lagrange Multipliers (a_n)
 - 5: Obtain the optimal direction and bias
 - 6: **end for**
 - 7: Obtain the decision boundary
-

Algorithm 2 (Backpropagation Algorithm) [23]

- Establish a feed-forward neural network
- 2: Initialize all the network weights
- for each** training example (x, t) **do**
- 4: Forward propagate the input through the network
 - a. Input the instance x into the network and calculate δ_h for each unit u in the network
- 6: Backward propagate the errors through the network
 - for each** network output unit k , the error term is computed as: **do**
 - 8: $\delta_k \leftarrow -o_k(1 - o_k)(t_k - o_k)$
 - end for**
 - 10: **for each** hidden unit h , the error term is calculated as: **do**
 - 12: $\delta_h \leftarrow o_h(1 - o_h) \sum_{k \in \text{outputs}} w_{kh} \delta_k$
 - 14: Update each network weight w_{jk} as follows:
 - 16: $w_{ji} \leftarrow w_{ji} + \Delta w_{ji}$
 - where;
 - $\Delta w_{ji} = \alpha \delta_j x_{ji}$
 - end for**

IV. RESULTS AND DISCUSSION

In this section, we present the effectiveness of the proposed sensor failure detection systems. We consider two different WSN deployment scenarios. Fig. 1 presents the single-hop

scenario, where the sensors can directly communicate with the base station. Fig. 2 presents a multi-hop scenario sensor deployment where the sensors communicate with the base stations through a router node.

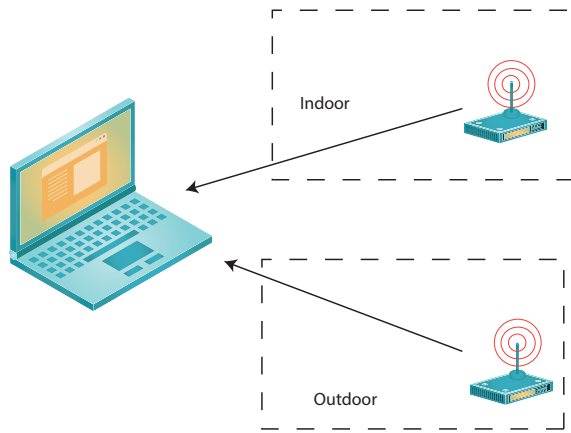


Fig. 1. Single-hop scenario sensor deployment [24].

A publicly accessible dataset of sensor measurements was trained using the single-layered NN and SVM machine learning methods with linear kernel function. Data from single-hop and multi-hop wireless sensor networks are included in the dataset, which is labeled wireless sensor networks data. Specifically, the dataset from four distinct sensors has been utilized to train the machine learning models to evaluate the performance of the sensor fault detections. The four different scenarios are the Indoor Single-hop scenario, Outdoor Single-hop scenario, Indoor multi-hop scenario, and Outdoor multi-hop scenario. In a single-hop scenario, there is direct communication between the sensor and the base station, sensors can directly send their sensed data to the base station. In a multi-hop scenario, the sensed data is routed through an intermediate node between the source and the destination. The dataset consists of two features which are the value of humidity and the value of temperature. These values are collected within six hours at an interval of 5 seconds. For both SVM and the single-layered perceptron, the data for all four sensors with different scenarios have been randomly shuffled and training utilizes 80% of the data, while the remaining 20% is allocated for testing. In perceptron, one hidden layer is used with a varying number of neurons. A varying number of neurons in the hidden layer will help in determining the best architecture with the smallest testing and training errors. Hence the architecture with the smallest testing and training errors will have the highest accuracy. Specifically, we have used 2, 4, 10, 20, 50, and 100 neurons in the perceptron hidden layer, and the average testing error

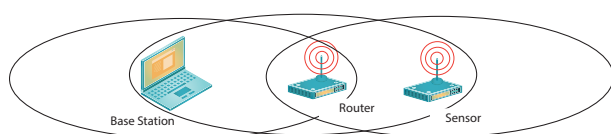


Fig. 2. Multi-hop scenario sensor deployment [24].

and training error have been observed.

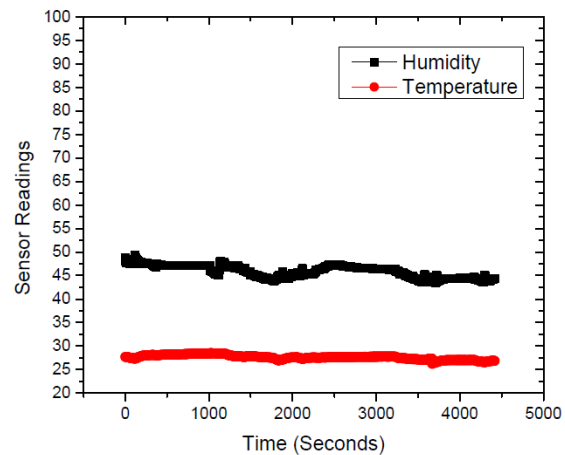


Fig. 3. Single hop scenario Indoor Sensor reading without anomalies

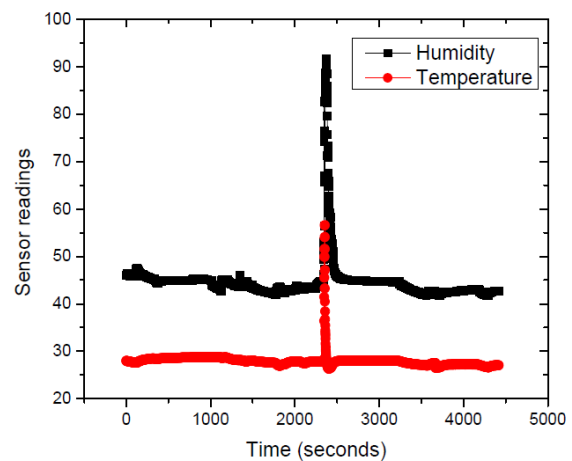


Fig. 4. Single hop scenario Indoor Sensor reading with anomalies

Fig. 3 presents the Indoor Sensor readings without faults for the Single hop scenario. It consists of humidity readings and temperature readings collected over six hours at an interval of every five seconds. Fig. 4 presents the Indoor Sensor reading with anomalies. From the Fig. 4, a spike can be observed at around halfway in the x-axis, this spike presents the anomalies in the sensor readings.

Figs. 5 to 8 present the results of the sensor faults detection using an SVM classifier. Measurements from four sensors have been used for the four different scenarios. Fig. 5 presents the SVM classification of the Indoor Sensor Single scenario. Fig. 6 presents the SVM classification of the Indoor Sensor Multi-hop scenario. Fig. 7 presents the SVM classification of the Outdoor single-hop scenario. Fig. 8 presents the SVM classification of the Outdoor multi-hop scenario. In the figures, the decision boundary of the SVM classification has separated the normal sensor data and the faulty sensor data. Normal sensor data is shown in red color and the faulty sensor data is shown in green color. Table 1 presents the Single-layered perceptron testing accuracy for sensor data of the four different scenarios.

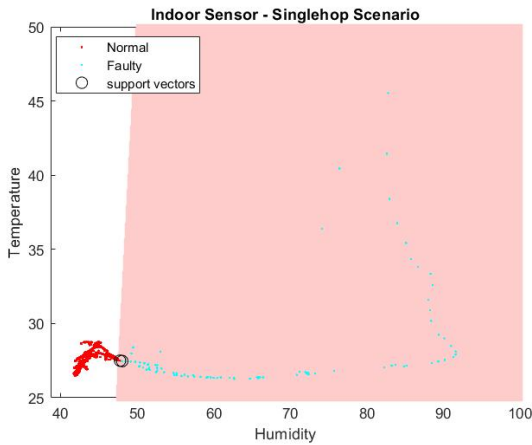


Fig. 5. SVM classification of Indoor Sensor Single-hop scenario

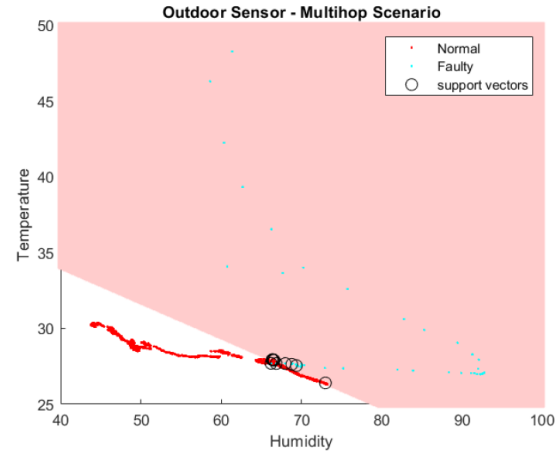


Fig. 8. SVM classification of Outdoor Sensor Multi-hop scenario

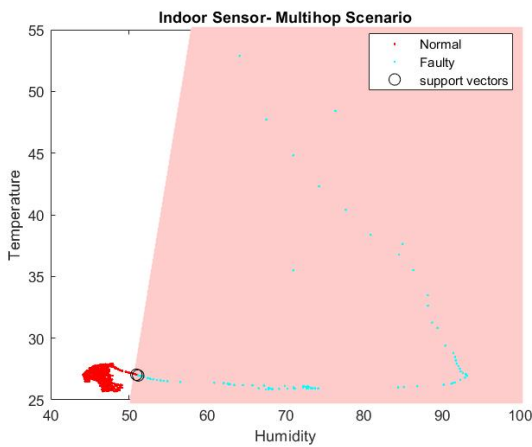


Fig. 6. SVM classification of Indoor Sensor Multi-hop scenario

Table 2 presents the performance comparison between the two machine learning methods used in sensor fault detection. In this study, testing the accuracy of the two machine learning methods is considered as the performance metric. The SVM classifier and the neural network, both two machine

learning methods have achieved good classification accuracy, as shown in the figure the accuracy of all four scenarios is above 99%. Moreover, SVM has higher accuracy compared to the neural networks for the three scenarios. Specifically, SVM has an accuracy of 99.77% for the indoor single-hop scenario compared to 99.38% accuracy of the neural network, SVM has an accuracy of 100.00% for the Indoor Multi-hop scenario compared to 99.77% accuracy of the neural networks, SVM has an accuracy of 99.89% for the Outdoor Multi-hop scenario compared to 99.44% accuracy of the neural networks, but for the Outdoor Single-hop scenario, the neural networks have higher classification accuracy of 99.98% compared to the SVM accuracy of 99.80%.

TABLE I. Testing Accuracy (%) for Different Hidden Layer Neurons

Hidden Layer Neurons	Indoor		Outdoor	
	Single-hop	Multihop	Single-hop	Multihop
2	99.98	99.98	99.99	99.99
4	99.98	99.99	99.99	99.98
10	99.99	99.99	99.99	99.99
20	99.99	100.00	99.99	99.99
50	99.88	99.76	99.99	99.99
100	96.48	98.90	99.95	96.69
Average	99.38	99.77	99.98	99.44

TABLE II. Scenarios Comparison: SVM vs. Single-layered Perceptron

Scenarios	SVM	Single-layered Perceptron
Indoor Single-hop	99.77%	99.38%
Indoor Multi-hop	100.00%	99.77%
Outdoor Single-hop	99.80%	99.98%
Outdoor Multi-hop	99.89%	99.44%

V. CONCLUSION AND FUTURE WORK

In this paper, machine learning-based sensor fault detection in WSN has been studied. Specifically, a performance comparison of SVM and Single-layered perceptron in sensor fault detection has been presented. The dataset used in this study is labeled wireless sensor networks data collected from a single hope scenario and multi-hop scenario from both indoor and outdoor sensor deployment. Specifically, four scenarios have been considered. The scenarios are indoor single-

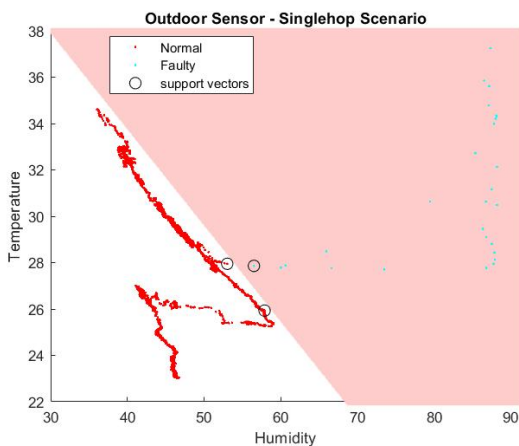


Fig. 7. SVM classification of Outdoor Sensor Single scenario

hop scenario, indoor multi-hop scenario, Outdoor single-hop scenario, and Outdoor multi-hop scenario. For all the aforementioned scenarios, the two machine learning methods have archived good classification accuracy of above 99%. From the simulation results, SVM has archived relatively higher accuracy than the Single-layered perceptron. In future work, we plan to use another sensor dataset that has more than two features and plan to use other machine learning algorithms and compare their performance so that we can observe the most suitable machine learning algorithm for the detection of sensor faults in WSN.

ACKNOWLEDGEMENT

This work was supported by the Faculty Development Competitive Research Grant Program of Nazarbayev University (grant no. 20122022FD4125).

REFERENCES

- [1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer networks*, vol. 52, no. 12, pp. 2292–2330, 2008.
- [2] T. Muhammed and R. A. Shaikh, "An analysis of fault detection strategies in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 78, pp. 267–287, 2017.
- [3] M. Yu, H. Mokhtar, and M. Merabti, "Fault management in wireless sensor networks," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 13–19, 2007.
- [4] Z. Feng, J. Q. Fu, and Y. Wang, "Weighted distributed fault detection for wireless sensor networks based on the distance," in *Proceedings of the 33rd Chinese Control Conference*. IEEE, 2014, pp. 322–326.
- [5] R. R. Panda, B. S. Gouda, and T. Panigrahi, "Efficient fault node detection algorithm for wireless sensor networks," in *2014 International Conference on High Performance Computing and Applications (ICHPCA)*. IEEE, 2014, pp. 1–5.
- [6] C. Titouna, M. Aliouat, and M. Gueroui, "Outlier detection approach using bayes classifiers in wireless sensor networks," *Wireless Personal Communications*, vol. 85, pp. 1009–1023, 2015.
- [7] A. Abid, A. Kachouri, A. B. F. Guiloufi, A. Mahfoudhi, N. Nasri, and M. Abid, "Centralized knn anomaly detector for wsn," in *2015 IEEE 12th International Multi-Conference on Systems, Signals & Devices (SSD15)*. IEEE, 2015, pp. 1–4.
- [8] S. Zidi, T. Moulahi, and B. Alaya, "Fault detection in wireless sensor networks through svm classifier," *IEEE Sensors Journal*, vol. 18, no. 1, pp. 340–347, 2017.
- [9] M.-H. Lee and Y.-H. Choi, "Fault detection of wireless sensor networks," *Computer Communications*, vol. 31, no. 14, pp. 3469–3475, 2008.
- [10] O. Obst, "Distributed fault detection in sensor networks using a recurrent neural network," *Neural processing letters*, vol. 40, pp. 261–273, 2014.
- [11] E. U. Warriach and K. Tei, "Fault detection in wireless sensor networks: A machine learning approach," in *2013 IEEE 16th International Conference on Computational Science and Engineering*. IEEE, 2013, pp. 758–765.
- [12] K. Zhang, K. Yang, S. Li, D. Jing, and H.-B. Chen, "Ann-based outlier detection for wireless sensor networks in smart buildings," *IEEE Access*, vol. 7, pp. 95 987–95 997, 2019.
- [13] F. Aliyu, S. Umar, and H. Al-Duwaish, "A survey of applications of artificial neural networks in wireless sensor networks," in *2019 8th International Conference on Modeling Simulation and Applied Optimization (ICMSAO)*, 2019, pp. 1–5.
- [14] P. Biswas, R. Charitha, S. Gavel, and A. S. Raghuvanshi, "Fault detection using hybrid of kf-elm for wireless sensor networks," in *2019 3rd international conference on trends in electronics and informatics (ICOEI)*. IEEE, 2019, pp. 746–750.
- [15] K. P. Sharma and T. P. Sharma, "rdfd: reactive distributed fault detection in wireless sensor networks," *Wireless Networks*, vol. 23, pp. 1145–1160, 2017.
- [16] A. Mahapatro and A. K. Panda, "Choice of detection parameters on fault detection in wireless sensor networks: A multiobjective optimization approach," *Wireless personal communications*, vol. 78, no. 1, pp. 649–669, 2014.
- [17] M. N. Sahoo and P. M. Khilar, "Diagnosis of wireless sensor networks in presence of permanent and intermittent faults," *Wireless personal communications*, vol. 78, no. 2, pp. 1571–1591, 2014.
- [18] —, "Distributed diagnosis of permanent and intermittent faults in wireless sensor networks," in *Advanced Computing, Networking and Informatics-Volume 2: Wireless Networks and Security Proceedings of the Second International Conference on Advanced Computing, Networking and Informatics (ICACNI-2014)*. Springer, 2014, pp. 133–141.
- [19] S. Gnanavel, M. Sreekrishna, V. Mani, G. Kumaran, R. Amshavalli, S. Alharbi, M. Maashi, O. I. Khalaf, G. M. Abdulsahib, A. D. Alghamdi *et al.*, "Analysis of fault classifiers to detect the faults and node failures in a wireless sensor network," *Electronics*, vol. 11, no. 10, p. 1609, 2022.
- [20] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.
- [21] B. Hou, B. Zhou, X. Li, L. Yi, Q. Wei, and R. Zhang, "Nonlinear error compensation of capacitive angular encoders based on improved particle swarm optimization support vector machines," *IEEE Access*, vol. 8, pp. 124 265–124 274, 2020.
- [22] T. M. Mitchell, *Machine learning*. McGraw-hill New York, 1997, vol. 1, no. 9.
- [23] A. Darii, M. Moll, M. S. Nistor, S. Pickl, O. C. Novac, C. M. Novac, I. M. Gordan, and C. E. Gordan, "Analysis, combination and integration of neuroevolution and backpropagation algorithms for gaming environment," in *2023 15th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*. IEEE, 2023, pp. 1–5.
- [24] S. Suthaharan, M. Alzahrani, S. Rajasegarar, C. Leckie, and M. Palaniswami, "Labelled data collection for anomaly detection in wireless sensor networks," in *2010 sixth international conference on intelligent sensors, sensor networks and information processing*. IEEE, 2010, pp. 269–274.



Abubakar Abdulkarim received his bachelor's degree of Science in electrical and electronic engineering at Eastern Mediterranean University, Turkish Republic of Northern Cyprus in 2017. He joined Binyaminu Usman Polytechnic Hadejia, Nigeria as an Assistant lecturer in 2019. He is currently a Master student in Electrical and Computer Engineering at Nazarbayev University, Kazakhstan. His research interests include Application of UAVs in wireless communication, Application of EV in logistic, Smart grid.



Israel Ehile Ehile received the B.Eng. in electrical and electronics engineering from the University of Agriculture, Makurdi, Nigeria in 2018. He was a Research Assistant with the Department of Electrical and Computer Engineering, Nazarbayev University, Astana, Kazakhstan in 2023. He is currently a Master student in Electrical and Computer Engineering at Nazarbayev University, Astana, Kazakhstan. His research interests include Communication in Inter-Satellite, Routing in Deep Space Network, Computer Networks and Cloud

Security.



Refik Caglar Kizilirmak (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2004 and 2006, respectively, and the Ph.D. degree from Keio University, Yokohama, Japan, in 2010. He was with the Communications and Spectrum Management Research Center, Turkey, on several telecommunication and defense industry projects. He is currently with the Department of Electrical and Computer Engineering, Nazarbayev University, Astana, Kazakhstan.

He has contributed to the technical requirements document of IEEE 802.15.7r1 standardization, which will enable visible light communication. He has authored several articles in the field of wireless communications and has filed three patent applications with the patent offices of USA and Japan.