




## RESEARCH ARTICLE

# Malicious Node Detection Using Machine Learning and Distributed Data Storage Using Blockchain in WSNs

MUHAMMAD NOUMAN<sup>1</sup>, UMAR QASIM<sup>2</sup>, HINA NASIR<sup>3,4</sup>,  
ABDULLAH ALMASOUD<sup>5</sup> , (Member, IEEE), MUHAMMAD IMRAN<sup>6</sup> , (Member, IEEE),  
AND NADEEM JAVAID<sup>1</sup> , (Senior Member, IEEE)

<sup>1</sup>Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan

<sup>2</sup>Department of Computer Science, University of Engineering and Technology Lahore (New Campus), Lahore 54000, Pakistan

<sup>3</sup>School of Electronic & Electrical Engineering, Institute of Robotics, Autonomous Systems and Sensing, University of Leeds, LS2 9JT Leeds, U.K.

<sup>4</sup>Department of Computer Science, Air University, Islamabad 44000, Pakistan

<sup>5</sup>Department of Electrical Engineering, College of Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

<sup>6</sup>Institute of Innovation, Science and Sustainability, Federation University, Brisbane, QLD 4000, Australia

Corresponding authors: Abdullah Almasoud (am.almasoud@psau.edu.sa) and Nadeem Javaid (nadeemjavaidqau@gmail.com)

This work was supported by the Deputyship for Research and Innovation, Ministry of Education, Saudi Arabia, under Project IF-PSAU-2021/01/19156.


**ABSTRACT** In the proposed work, blockchain is implemented on the Base Stations (BSs) and Cluster Heads (CHs) to register the nodes using their credentials and also to tackle various security issues. Moreover, a Machine Learning (ML) classifier, termed as Histogram Gradient Boost (HGB), is employed on the BSs to classify the nodes as malicious or legitimate. In case, the node is found to be malicious, its registration is revoked from the network. Whereas, if a node is found to be legitimate, then its data is stored in an Interplanetary File System (IPFS). IPFS stores the data in the form of chunks and generates hash for the data, which is then stored in blockchain. In addition, Verifiable Byzantine Fault Tolerance (VBFT) is used instead of Proof of Work (PoW) to perform consensus and validate transactions. Also, extensive simulations are performed using the Wireless Sensor Network (WSN) dataset, referred as WSN-DS. The proposed model is evaluated both on the original dataset and the balanced dataset. Furthermore, HGB is compared with other existing classifiers, Adaptive Boost (AdaBoost), Gradient Boost (GB), Linear Discriminant Analysis (LDA), Extreme Gradient Boost (XGB) and ridge, using different performance metrics like accuracy, precision, recall, micro-F1 score and macro-F1 score. The performance evaluation of HGB shows that it outperforms GB, AdaBoost, LDA, XGB and Ridge by 2-4%, 8-10%, 12-14%, 3-5% and 14-16%, respectively. Moreover, the results with balanced dataset are better than those with original dataset. Also, VBFT performs 20-30% better than PoW. Overall, the proposed model performs efficiently in terms of malicious node detection and secure data storage.

**INDEX TERMS** Blockchain, histogram gradient boost, IPFS, malicious node detection, VBFT, WSN.

## I. INTRODUCTION

A Wireless Sensor Network (WSN), comprising thousands of nodes, is widely used in several applications like supply chain management, military surveillance, environmental

monitoring, etc., [1]. Sensor Nodes (SNs) are used to monitor and gather environmental data. Besides, in crowd sensing networks, SNs send massive amounts of the collected data to the nearby nodes and Cluster Heads (CHs). This process decreases the cost of different types of equipment and conventional methods for data collection. However, some nodes do not participate in crowd sensing networks due to privacy issues.

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

Moreover, in the absence of a security mechanism, WSNs become vulnerable to malicious nodes that modify the data for their own interest. Furthermore, the SNs are resource constrained and do not perform efficient resource utilization. In addition, traditional methods are unable to detect malicious nodes. Whenever an attack is performed by a malicious node, the network is compromised, and malicious nodes perform malicious activities that affect the entire network. To prevent the nodes from acting maliciously, many authors propose authentication schemes that allow only the authentic nodes to join the network [2]. However, the existing authentication schemes depend upon centralized entities, which are vulnerable to cyber-attacks.

In WSNs, SNs are either randomly or statically deployed depending upon network topology. SNs gather environmental data and transfer it to their destination. However, some SNs do not store the location information because their topology is frequently changed, and the usage of a large number of sensing nodes may cause network information congestion. To solve this issue, a WSN is split into sub-networks that CHs manage. CHs get data from SNs and send it to Base Stations (BSs) [3]. Moreover, SNs are resource constrained in terms of low storage and computational power. Also, SNs are prone to different types of attacks and are easily compromised by malicious nodes. Many researchers propose different techniques to avoid malicious attacks and detect malicious nodes [4], [5]. However, detection of the malicious node in WSNs depends on a third party, which can easily be compromised. Therefore, blockchain is introduced to overcome the problems associated with centralization and the involvement of third parties [6], [7], [8], [9], [10].

WSN nodes produce vast amount of data and store them on a centralized system. However, security breaches and failures might destabilize the WSNs. Therefore, a Peer-to-Peer (P2P) network is proposed to overcome centralization issues related to data storage [11]. In a P2P network, nodes directly transfer the data from the source to the destination without the assistance of a third party. With the rapid expansion of WSN nodes, P2P architecture faces security and privacy challenges. Therefore, blockchain technology is introduced to address the security issues of WSNs through a distributed, decentralized, and immutable ledger [12]. Once data is added to the blockchain, it will never be tampered by any malicious party due to the distributive nature of the blockchain. Furthermore, the idea of integrating WSN and blockchain has attracted much attention from the public. However, blockchain consumes a lot of computational resources, whereas, SNs have limited resources. Also, when incorporating the new blockchain design into the WSNs, some other issues may arise. Besides, the Proof of Work (PoW) consensus mechanism is widely used in blockchain that effectively reduces the number of malicious nodes and verifies the transaction. However, the PoW consensus mechanism requires a large amount of computational power to confirm a transaction and add it to the block in the blockchain [13].

Moreover, most of the researchers propose the Interplanetary File System (IPFS) for data storage, which was introduced by Juan Benet [14]. IPFS shares many of the same characteristics as blockchain. It uses a P2P, decentralized, and distributed file storage system. Besides, IPFS nodes are the machines that execute the IPFS software to store and retrieve files from the IPFS network. IPFS nodes use content addressing to store and retrieve the files. All IPFS nodes store the files in the form of chunks, similar to a BitTorrent network. There is no effect on the network if one node fails. Furthermore, it uses two types of data structures to distribute the file. One is Distributed Hash Table (DHT), and the second is Merkle Directed Acyclic Graph (DAG). When nodes send a file to the IPFS for storage, then SHA256 algorithm is executed on the file, and the hash value for each stored file is generated. The hash value is called a content identifier, which is used to retrieve the stored files from the IPFS.

### A. MOTIVATION AND CONTRIBUTIONS

The motivation of this work comes from the fact that in WSNs, nodes are randomly deployed. This random deployment leads to various issues like loss of data, security risk, etc. In WSNs, data is collected from the surrounding environment. WSNs are easily accessible, and any node can join them. As a result, malicious nodes enter the network and perform malicious activities that affect the entire network. The authors in [15] propose a centralized authentication mechanism that registers the nodes and protects confidential node identification from an unauthorized node in WSNs. However, the centralized system causes the issue of a single point of failure. Moreover, SNs have resource constraints and do not efficiently detect malicious behavior in the network. Also, malicious nodes can easily damage and compromise the WSNs [16]. Furthermore, malicious nodes collect false data and deliver it to destination nodes where blockchain is deployed to store the data [17]. However, storing huge volumes of data in a blockchain is very expensive. In addition, blockchain uses the PoW consensus mechanism for block generation, which consumes a huge amount of computation power during block generation [13]. Further motivation can be taken from [18]. The results are provided in Section IV in the manuscript.

The proposed model's key contributions include the following:

- in a WSN, a blockchain based decentralized authentication mechanism is used to protect disclosure of node identities by external nodes,
- for data storage in a WSN, IPFS is deployed that integrates blockchain technology. The cost of storing data in the blockchain is minimized when storing data on IPFS. The data is stored in chunks in IPFS, and the hashes are created that are recorded in the blockchain,
- the proposed blockchain based network uses the Verifiable Byzantine Fault Tolerance (VBFT) consensus mechanism [19], which reduces the blockchain

transaction cost and increases the throughput as compared to the existing consensus mechanisms like PoW and

- the comparative analysis of the proposed classifier, i.e., Histogram Gradient Boost (HGB), with Adaptive Boost (AdaBoost), Gradient Boost (GB), Extreme Gradient Boost (XGB), Linear Discriminant Analysis (LDA), and ridge classifiers is performed. The analysis is done on the basis of numerous performance metrics, including accuracy, precision, recall, micro F1-score, and macro F1-score.

The remainder of the manuscript is organized as follows. The related work is presented in Section II, while the problem statement and proposed system model are given in Section III and Section IV, respectively. Section VI presents the outcomes of the simulations performed to verify the accuracy of the proposed model. Section VI provides the feasibility of the proposed model. In Section 7, the conclusion of the manuscript is provided.

## II. RELATED WORK

In WSNs, SNs share information and communicate with each other. WSNs are easily accessible, and any node can join them. Malicious nodes acquire legitimate node identities, which makes it easy for them to become part of the network. The authors propose a lightweight blockchain IoT authentication scheme in [20]. This scheme ensures integrity and non-repudiation in the network. Whenever IoT nodes communicate with one another, they must first authenticate each other, which is done using a lightweight blockchain. In [21], the authors develop a hybrid blockchain model for IoT nodes to prevent malicious or fake data packets from spreading throughout the network. Public and private blockchain make up a hybrid blockchain. Between CHs and BSs, the public blockchain is implemented, while the private blockchain is implemented between CHs and SNs. SNs are authenticated on CH using a smart contract, and CHs are authenticated on BS. In [22], blockchain and reinforcement learning based model is proposed for efficient and secure routing in WSNs. The reinforcement learning algorithm selects the best possible routing path. It avoids the malicious routing links that might send data through compromised nodes, while blockchain is used for node authentication and managing all routing information. In [23], blockchain based key management is presented to tackle the issue of certificate-less key management. The blockchain performs node authentication, registration, and joining or quitting of nodes. In addition, it provides the mechanism for the detection of the compromised node. In [24], a data structure based on blockchain is used to hold nodes' authentication and trust information. Blockchain authentication consists of three aspects: public keys, block mining, and mutual influence, while the blockchain trust model consists of two aspects: knowledge based trust and trust evaluation. In [25], blockchain is used to overcome IoT issues. IoT devices register themselves on

the blockchain. If the IoT device is successfully registered and authenticated, the activity is performed according to its capability. Similarly, users need to be authenticated in the blockchain network to be able to control and manage IoT devices. It restricts the malicious nodes from becoming a part of the network and stores all evidences on a blockchain. In [26], the modified version of the station-to-station (STS) protocol is presented. It first authenticates the user and then establishes a secret exchange session key that ensures user anonymity inside a group.

In [29], a blockchain based data structure model is used for malicious node detection. WSN nodes have limited memory and computational power, and are unable to detect malicious nodes. Whenever an attack is performed on a node, it is compromised by a malicious node. In [30], the authors propose the three layered SDN architecture that monitors and analyzes the traffic in the IoT environment. Another pertinent point is that a blockchain is used for decentralized attack detection. As a result, fog computing and mobile edge computing provide attack detection, reducing the number of attacks that occur at the edge layer. In [31], a secure and privacy-preserving model is proposed for the smart city. Three modules make up the proposed model. The first module is trustworthiness, where authors use the blockchain among the IoT devices to maintain trust. The second module is two-level privacy, where enhanced PoW is used in blockchain to achieve confidentiality and prevent the poisoning attack. The third module is the intrusion detection system, which is used for malicious node detection. XGB classifier is utilized in the process of identifying malicious nodes. In [32], the authors propose the secure privacy-preserving framework. The presented model has two major components: two-level of privacy and an intrusion detection mechanism. Blockchain is utilized in two-level privacy to securely transmit data among IoT nodes. The two-level privacy uses principal component analysis (PCA) to transform data into a new form to protect it against inference attacks. The authors use gradient boost anomaly detection (GBAD) for the intrusion detection system based on light gradient boost model (LGBM). GBAD is deployed in a smart city that can proficiently classify normal and malicious observations. In [33], a blockchain based automatic (AutoML) model is proposed for customer services to overcome the third parties' challenges. IoT devices are used to collect data, and blockchain is used for secure data exchange in an open environment. Furthermore, AutoML is designed to process data and reduce expert costs. In [34], the authors propose an ensemble learning technique that uses multiple ML techniques to classify data. The final classification report is obtained based on all classifiers' votes.

In [35], secure routing with multi-layered IoT architecture is proposed, where light blockchain and cloud are used. Light blockchain is used for security and privacy, while the cloud is used for data storage. In [36], two different kinds of blockchain are used in a WSN: one for storing data and the other for managing how users can access data. A verifiable

TABLE 1. Summarized related work table.

Limitations Already Addressed	Solutions Already Proposed	Validations Already Done	Limitations to be Addressed
Malicious nodes are present in the network [20]	Intrusion prevention framework is proposed	Data integrity and network connectivity	XoR function is not strong enough, blackhole and greyhole attacks may occur
Localization problem of unknown nodes [21]	Trust model based on blockchain is used	Feasibility, fairness and traceability	RSA slows down the encryption method in case of large data
Crowd sensing networks are vulnerable [22]	Confusion mechanism and blockchain based incentive mechanism are proposed	Energy consumption, delay	Not improved in route acquisition latency and packet delivery ratio
Centralization method is used for registration [23]	A hybrid blockchain based model is proposed	Processing time and transmission delay	Data duplication
Localization of WSN nodes, unknown nodes perform attacks on network [24]	A blockchain trust model is proposed	False negative rate, detection accuracy and energy consumption	No encryption and hashing algorithm are used for security
Dynamically routing and centralized registration [25]	Blockchain and reinforcement learning algorithm are used	Delay, energy consumption	Queue delay and processing delay
IoT node manufacturers are unable to agree on a simple central administrator [26]	BCR protocol is proposed	Packet drop ratio, packet delivery ratio, delay	Not improved in route acquisition latency and packet delivery ratio
Balance energy consumption of sensor nodes and to improve WSN longevity [27]	Dynamic hierarchical protocol based on combinatorial optimization is proposed	A hierarchy-based connection mechanism to construct a hierarchical network structure	Processing delay and computational complexity
Reduced lifetime of ultra-dense WSNs [28]	Unsupervised learning approach	Residual energy and computational complexity	Increased computational complexity

data possession consensus mechanism is also used to reduce the cost of computing. In [37], a decentralized blockchain mechanism is proposed for Internet of Things (IoT) monitoring and controlling, in which each entity can track and communicate. The data controller manager receives the data and filters out specific data stored in the blockchain. In [38], the multiple synergistic proofs green consensus method is proposed to address the issue of limited data storage. As a result, less space is being allocated to blockchain data. When peer nodes verify a transaction or a block, they frequently send the same information. This is not favourable and deteriorates network performance. In [39], a blockchain based aggregation scheme is proposed that decreases the device's duty cycle. Moreover, the reduction in the risk of transmitting a large amount of data at the risk of increasing data delay at the IoT device is achieved. The selection of gathered data is based on the channel's quality, and the most recent data structure statistics. The expense is incurred in sending the Merkle-Patricia tree data structures as evidence of inclusion for the most recent data. In [40], an optimized sampling rate strategy is proposed for IoT sensors that transmit data using blockchain and Tangle technologies, which decrease the age of information, and make efficient end-user processing and networking resources.

In [41], blockchain is used to resolve IoT networks' security and privacy issues, providing a secure distributed and immutable ledger. Some nodes in a blockchain network are responsible for mining, which entails validating transactions and adding new blocks to the blockchain. Mining nodes require a high computing capacity as compared to conventional nodes. IoT devices have limited power, battery life, etc. This research encourages IoT devices to purchase processing power through edge servers, participate in mining, and

earn incentives on the blockchain network. In [42], a scalable and secure blockchain is proposed for the IIoT system. First, a self-adaptive PoW consensus mechanism is proposed, which adjusts the difficulty for nodes as per their behaviours. The self-adaptive PoW consensus mechanism efficiently reduces computational power. Moreover, asymmetric cryptography is used for access control, giving users more options for managing data authority. Furthermore, a blockchain based directed acyclic network is used to improve throughput and transaction time. In [43], a hybrid model based on an SDN and blockchain is proposed for the smart city. Two types of smart city nodes exist in the network: edge node and core node. The core nodes are provided with the data from the edge nodes after the edge nodes receive the data from the sensors. These edge nodes act as centralized entities because edge nodes also use SDN technology, and their computing power and storage are less than that of core nodes. Whereas, core nodes are the powerful nodes that receive the sensors' data and perform mining. Core nodes, also called miner nodes, use blockchain technology for mining transactions and enhancing security.

In [44], a lightweight exclusive OR (XOR) hash algorithm is used, which provides secure and reliable data routing using blockchain technology. In [45], rolling blockchain is used for WSNs to ensure that the WSN nodes and data are secured from attackers. In [46], a blockchain system with mobile edge computing allows mobile miner nodes to perform computationally intensive tasks on surrounding edge nodes. As a result of this strategy, backhaul and latency are minimized. In [47], a trust-aware localization routing protocol with class based dynamic encryption is proposed. This proposed method first searches the secure path from source to destination and then forwards the data packet. The selection

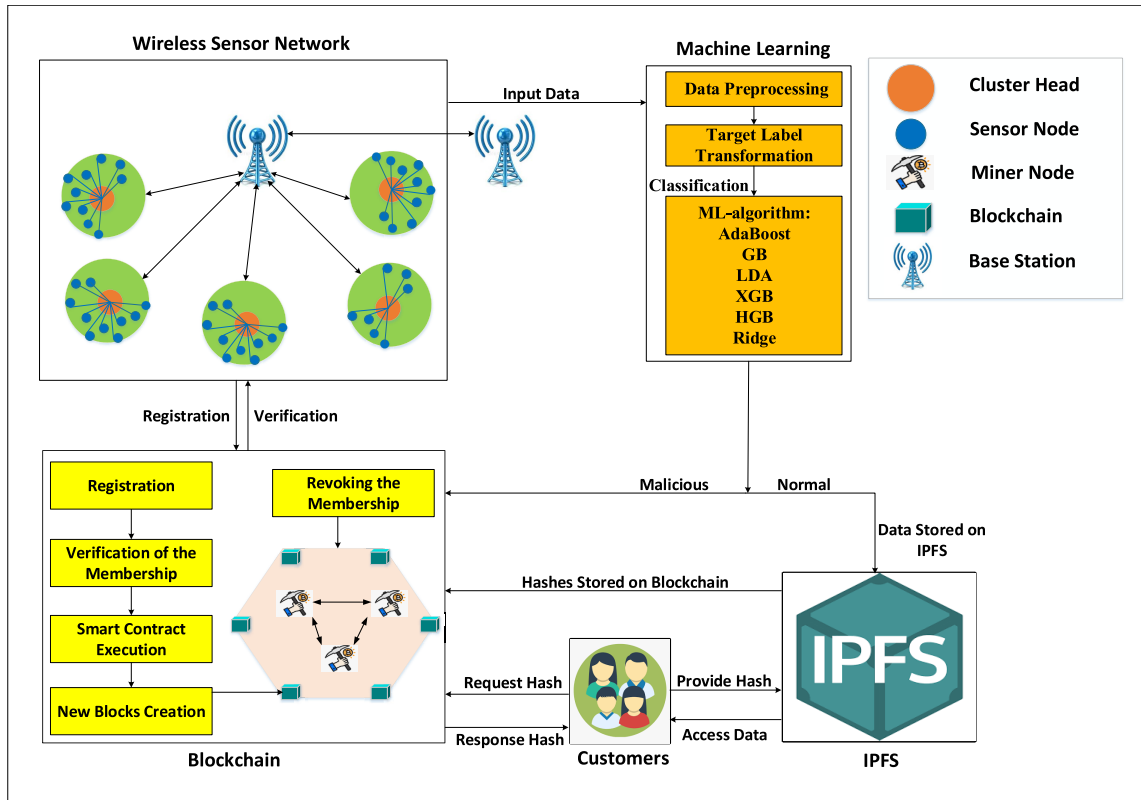


FIGURE 1. Proposed network model for WSNs.

of a secure path is made using the trust value. Moreover, blockchain based encryption is used for data integrity. In [48], a trust based range-free safe localization method is proposed. The trust values of beacon nodes are communicated through the blockchain with their nearby nodes. Trustworthy beacon nodes are selected as miners for block mining. However, it is a time consuming process. Table 1 presents the summarized related work.

### III. PROPOSED SYSTEM MODEL

In this section, the WSN network, developed in the proposed work, is discussed. Also, this section introduces the reasonable assumptions that are used to propose the network for WSNs. Figure 1 presents the proposed system model.

#### A. ASSUMPTIONS

- SNs and CHs are resource constrained and each node has a unique identity.
- BSs provide a certain amount of data storage and computational power for processing the data sent by the SNs.
- BSs are resource enriched and trustworthy nodes for the CHs and SNs.
- There is a possibility of the occurrence of energy nodes in WSNs, as discussed in [49]. However, in the proposed work, it is assumed that no energy holes exist.

#### B. FORMULATION PROBLEM

In WSNs, nodes are randomly deployed, and data is collected from the surrounding environment. WSNs are easily accessible, and any node can join them. As a result, malicious nodes enter the network and perform malicious activities that affect the entire network. The authors in [15] propose a centralized authentication mechanism that registers the nodes and protects confidential node identification from an unauthorized node in WSNs. However, the centralized system causes the issue of a single point of failure. Moreover, SNs have resource constraints and do not efficiently detect malicious behaviour in the network. Also, malicious nodes can easily damage and compromise the WSNs [16]. Furthermore, malicious nodes collect false data and deliver it to destination nodes where blockchain is deployed to store the data [17]. However, storing huge volumes of data in a blockchain is very expensive. In addition, blockchain uses the PoW consensus mechanism for block generation, which consumes a huge amount of computation power during block generation [13].

#### C. SYSTEM MODEL DESCRIPTION

Our proposed network model's nodes are divided into SNs, CHs, and BSs. SNs are randomly deployed. Whereas, CHs are chosen based on their high residual energy relative to SNs and their closeness to the BSs. SNs and CHs are registered

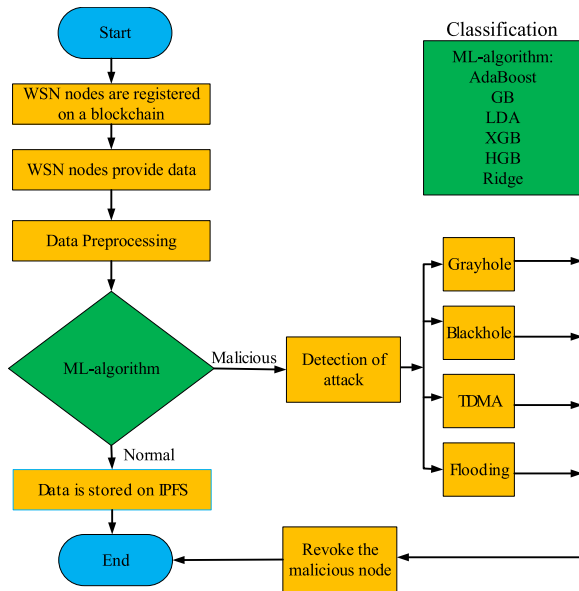


FIGURE 2. Proposed model's workflow.

on the blockchain, which is deployed on BSs. Following the registration, the blockchain authenticates SNs and CHs using Node\_ID. SNs collect data and send it to CHs. While CHs process data and transmit it to BSs. BSs utilize an ML classifier, HGB, to determine whether data is transmitted to a malicious node or a normal node. When data is transferred to a malicious node, the HGB classifier quickly recognizes that the data belongs to which malicious class, classifies the attack to that class, and reports to the blockchain. Blockchain then revokes the malicious node's registration. Otherwise, data is stored in an IPFS database. IPFS generates a unique identifier (hash) for the data and sends it back to the BSs, where it is stored on the blockchain. Moreover, public blockchain is implemented on the BSs. The public blockchain is customized to allow nodes to add, remove and validate transactions. Furthermore, a VBFT consensus mechanism is used in the blockchain to verify and store transaction nodes' credentials and cryptography hashes [19]. Figure 2 is a representation of the workflow associated with the proposed model. The steps involved in the system model are given in Algorithm 1.

#### Algorithm 1 Pseudo Code of the Proposed Model

**Step-1:** All nodes are registered on the BSs, where blockchain is implemented.

**Step-2:** SNs collect data from the surrounding area and relay it to CHs, while CHs forward the data to the BSs [64].

**Step-3:** The BSs are trained on an ML classifier, HGB, that classifies the data and sends it either to a malicious node or a normal node.

**Step-4:** Data is stored on the IPFS if the BS classifies the node as a normal node. Otherwise, the BS revokes the node's registration.

**Step-5:** The IPFS provides hashes that are stored on the blockchain implemented on the BSs.

#### D. REGISTRATION

SNs and CHs play an important part in the overall system model. SNs and CHs must create accounts on the blockchain and send registration requests to BSs. After getting their registration responses, the SNs and CHs send data to BSs.

#### E. SENSOR NODES

In the proposed model, we adopted the WSN model, which is also used by [64]. SNs are randomly deployed according to their functionalities. SNs collect data from their surrounding area and transmit it to the CH node. Each SN is directly connected to a CH in the network and shares its data and credentials after it is fully registered. The credentials are stored on the blockchain, which ensures that the SNs are secured. The data once stored in the blockchain can not be forged as blockchain becomes data integrity and tamper-proof nature. The security of SNs is necessary because if they are secured, only then the data coming from them can be regarded as authentic, as in [65].

#### F. CLUSTER HEADS

CHs are intermediate nodes that receive data from SNs, process it, and then pass it to the BSs. CHs' storage capacity and computational power are higher than those of SNs and lower than those of BSs. Each CH is directly connected to a BS to shares its information and credentials with it.

#### G. BASE STATION

A BS is a powerful node that has the highest computational power and storage capacity in the proposed network. It is also considered the core node of a network. In the proposed network, BSs receive data from CHs, perform some complex operations, and verify if the data is being transferred to a malicious node or a normal node. The malicious node's registration is revoked if data is delivered to it. If not, it keeps information in the IPFS database. BSs store the network nodes' credentials and monitor the whole network. BSs serve as trusted nodes for other nodes or subnetworks in the network.

#### H. CUSTOMERS

In the proposed model, shown in Figure 1, customers rely on IPFS and blockchain. Customers need to access the data that the SNs gathered. Therefore, customers are initially registered on the blockchain. The blockchain confirms whether the customer identity exists or not. If the customer is validated, it is allowed to join the network. A customer enters the network and requests the hash of the desired data, already recorded on the blockchain. The customer provides the hash value to the IPFS database to retrieve the data associated with the hash.

#### I. MALICIOUS NODE DETECTION USING MACHINE LEARNING CLASSIFIERS

Some malicious nodes enter the network as legitimate nodes to complete their registration process. After the registration,

these malicious nodes change their behaviour and act maliciously to attack the network nodes. In our proposed system, the ML classifier is deployed on BSs to classify the normal and malicious nodes. We conduct comparative analyses using six different ML classifiers for malicious node detection. These classifiers are AdaBoost, GB, XGB, LDA, ridge, and HGB, which are used to classify malicious and legitimate nodes. After nodes' classification, BSs revoke the malicious nodes from the network. The classifiers are further discussed below.

Freund and Schapire invented AdaBoost in 1996 [50]. It was the first ensemble boosting technique, which aims to combine multiple weak learners and make a strong model because a single weak learner is not able to predict an accurate class. The combination of weak classifiers makes a new strong classifier after the voting mechanism, and AdaBoost is one of them. Because of less time complexity, fast performance, and no difficulty in implementation, AdaBoost is the most efficient and effectively used classifier in computer vision. Also, boosting methods are considered greedy in terms of dealing with the exponential error function. The usage of AdaBoost improves the accuracy of weak classifiers. This algorithm initially assigns equal weights to all samples and passes them to the first weak learner. The weak classifier is trained, giving the output in the form of 1, -1. After that, weights are assigned in the second round to each observation. This process is repeated several times, creating a set of weak classifiers. The time complexity of AdaBoost is  $O(\text{ftn})$  [51]. Here,  $f$  represents the features,  $t$  represents the weak learners while the number of dataset samples is presented by  $n$ . The AdaBoost algorithm is presented below.

---

#### Algorithm 2 AdaBoost Algorithm

---

Initialize the observation weights  $w_i = 1/N$ ,  $i = 1, 2, \dots, N$ .  
**for**  $m=1$  to  $M$

(a) Fit a classifier  $G_m(x)$  to training data using weight  $w_i$ .

(b) Compute  $err_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}$ .

(c) Compute  $\alpha_m = \log((1 - err_m)/err_m)$ .

(d) Set  $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m)]$ ,  $i = 1, 2, \dots, N$ .

**endfor**

Output  $G_m(x) = \text{sign}[\sum_{m=1}^M \alpha_m G_m(x)]$ .

---

GB is a supervised ML algorithm invented by Friedman, 2001 [52]. It is an ensemble technique that is used for classification and regression. It is different from AdaBoost, and it comprises three parts: loss function, weak learners, and additive model. The loss function is used to minimize the residual and converge the final output. While the weak learners are used to make predictions. Initially, GB uses two models to start with a base model and find the residual passes to the first weak learner. It combines several weak learners and makes a single strong learner using the additive model. Individual weak learners act as decision trees (DT) in GB. These DTs are constructed so that each new tree fits within the residuals of the previous step, allowing the model to reduce error. A new

DT learned from the mistakes of a prior DT. These DTs are sequentially connected to each other, and each DT minimizes the error of the previous DT. Furthermore, the additive model combines the outcomes of each step, given the strong learner. The time complexity of GB is  $O(\text{ftn} \log n)$  [53]. Here,  $\log n$  represents the depth of the weak learners. The GB algorithm is presented below.

---

#### Algorithm 3 Gradient Boost Algorithm

---

Initialize model with constant value:  $F_{0,x} = \text{argmin}_r \sum_{i=1}^n L(y_i, r)$ .

**for**  $m = 1$  to  $M$

Compute residual  $r_{im} = -[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}]_{F(x)=F_{m-1}(x)}$  for  $i = 1, \dots, n$

Train regression tree with feature  $x$  against  $r$  and create terminal nodes reasons  $R_{jm}$  for  $j = 1, \dots, m$

Compute  $r_{jm} = \text{argmin}_r \sum_{x_i \in R_{jm}} L(y_i, F_{(m-1)(x_i)} + r)$  for  $j = 1, \dots, m$

Update the model:  $F_m(x) = F_{m-1}(x) + v \sum_{j=1}^{j_m} r_{jm}$  ( $x \in R_{jm}$ )

**endfor**

Output  $f(x) = F_m(x)$ .

---

LDA is a broader variant of Fisher discriminate analysis (FDA), also known as normal discriminate. LDA is a supervised ML technique used for classification and dimensionality reduction, invented by Ronald A. Fisher in 1936 [54]. The primary goal of LDA is to reduce higher dimension data into lower dimension data to prevent losing important information and reduce the consumption of computational resources. However, the number of features surpassing the number of samples along with the nonlinearity of the data points cause the LDA to fail. In dimensionality reduction, three steps are involved. In the first step, separability between the classes, known as between the class matrix or between the class variance, is calculated. The goal is to maximize the separation between the two classes. The difference between the mean of class and the data point of a class, known as within class matrix or within class variance, is calculated in the second step. The aim of this calculation is minimizing the within class matrix or within class variance. In the third step, the new lower dimensional space is built and the new data point is projected onto it. The time complexity of LDA is  $O(Mf^2)$  [55] in the case where the number of instances exceeds the number of features.  $M$  represents the mean of instances. The LDA algorithm is presented below.

XGB was created by Tianqi Chen in 2014 in order to improve the performance and speed of ML models [56]. Despite being scalable and a highly accurate extension of GB for boosted tree algorithms, XGB needs heaps of computing power. It refers to engineers' goal of pushing the limit of computation resources for the GB technique. This technique sequentially generates DTs. Weights are very important in XGB. All independent variable are assigned weights and subsequently fed into the DT predicting results. When the

**Algorithm 4** Linear Discriminant Analysis Algorithm

Given a set of  $N$  samples  $[x_i]_{i=1}^N$ .  
 Compute the mean of each class  $\mu(1 \times M)$ .  
 Compute the total mean of all data  $\mu(1 \times M)$ .  
 Calculate between class matrix  $S_B(M \times M)$ .  
**for** Class  $i=1, 2, \dots, c$   
   Compute within-class matrix of each class  $S_{w_i}(M \times M)$ ,  
 as follows:  
    $S_{w_i} = \sum_{x_i \in w_j} (x_i - \mu)(x_i - \mu)^T$   
   Construct a transformation matrix for each class ( $W_i$ ) as  
 follows:  
    $W_i = S_{w_i}^{-1} S_B$   
   The eigenvalues ( $V$ ) and eigenvectors ( $\omega$ ) of each transfor-  
 mation matrix ( $W_i$ ) are calculated, the calculated eigenvector  
 and eigenvalues of the  $i$ -th class.  
   Sort the eigenvectors in descending order according to their  
 corresponding eigenvalues.  
   Project the samples of each class onto their lower dimen-  
 sional space.  
    $Y=X.T$   
**endfor**

tree wrongly predicts a variable, the weight of the variables is increased, and these variables are provided in the second DT. These various predictors are combined to form a more robust and precise model. Three steps are performed in XGB. Firstly, it reduces overfitting by using regularization. Second, it optimizes sorting with parallel execution, which increases runtime speed. Finally, it prunes the tree using the maximum depth of the DT as a parameter, minimizing the total runtime. The time complexity of XGB is  $O(\text{tdxlogn})$  [57]. Here,  $d$  represents the height of the tree, and  $x$  represents the missing values. The XGB algorithm is presented below.

**Algorithm 5** Extreme Gradient Boost Algorithm

**Data:** Dataset and hyperparameters Initialize  $f_0x$ ;  
**for**  $k=1$  to  $M$     Calculate  $g_k = \frac{\partial L(y,f)}{\partial f}$ ;  
   Calculate  $h_k = \frac{\partial^2 L(y,f)}{\partial f^2}$ ;  
 Determine the structure by choosing splits with maximized  
 gain  
    $A = \frac{1}{2} [\frac{G_L^2}{H_L} + \frac{G_R^2}{H_R} + \frac{G^2}{H}]$ ;  
   Determine the leaf weights  $w^* = -\frac{G}{H}$ ;  
   Determine the base learner  $b(x) = \sum_{j=1}^T w_j I$ ;  
   Add trees  $f_k(x) = f_{k-1}(x) + b(x)$ ;  
**endfor**  
 Result  $f(x) = \sum_{k=0}^M f_k(x)$

A histogram is used to count the frequency of data across a specific time period. It is also known as binning or bucket [58]. Instead of calculating the split points on the sorted feature values, HGB applies the binning method to the DT [59]. The binning method is applied to data for pre-processing, which sorts the feature values and then divides the sorted feature values into numerous buckets or bins. It makes

this algorithm more efficient as compared to XGB, LGB, and GB in terms of memory consumption and training speed. It is also used to convert continuous or numerical variables into categorical features and deal with noisy data. The time complexity of HGB is  $O(\text{ft}(n\_bins))$  [60]. Here,  $n\_bins$  represents the data instances in the current block to generate the histogram. The HGB algorithm is presented below.

**Algorithm 6** Histogram Gradient Boost Algorithm

Initialize model with constant value:  $F_0x = \text{argmin}_r \sum_{i=1}^n L(y_i, r)$ .  
**for**  $m=1$  to  $M$   
 Compute residual  $r_{im} = -[\frac{\partial L(y_i, F(x))}{\partial F(x)}]_{F(x)=F_{m-1}(x)}$   
 for  $i=1, \dots, n$   
   Apply binning technique.  
   Sort the data features.  
   Distribute the data feature in bins.  
   Train regression tree with feature  $x$  against  $r$  and create  
 terminal nodes region  $R_{jm}$  for  $j=1, \dots, m$   
   Compute  $r_{jm} = \text{argmin}_r \sum_{x_i \in R_{jm}} L(y_i, F(m-1)(x_i) + r)$   
 for  $j=1, \dots, m$   
   Update the model:  $F_m(x) = F_{m-1}(x) + v \sum_{j=1}^m r_{jm} 1$   
 ( $x \in R_{jm}$ )  
**endfor**  
 Output  $f(x) = F_m(x)$ .

A ridge classifier is a type of ridge regressor. The ridge classifier first converts the target variable into binary form (1, -1) and then treats it as a ridge regressor. Hoerl and Kennard introduced the ridge regressor in 1970 as a regularization method for reducing model complexity [61], [62]. The time complexity of the ridge classifier is  $O(n^3)$  [63]. It uses the coefficient estimator for variables that are not linearly independent but are highly correlated. The ridge estimator shrinks the coefficient value and produces a new value close to the actual population. Furthermore, it involves plenty of coefficient mechanisms, meaning that no coefficient is left when the model is built. Due to the penalty mechanism, the loss function is minimized. The ridge classification algorithm is presented below.

**Algorithm 7** Ridge Classification Algorithm

**Step-1:** Input data matrix  $X$  holds training dataset and data matrix  $X$ -test holds the test dataset.  
**Step-2:** For each test data  $x \in X$ -test, calculate the regression parameter vector  $\hat{\alpha}$  as  $\hat{\alpha} = \text{argmin}_\alpha \|x - X_i \alpha\|_2^2 + \lambda \|\alpha\|_2^2$  where,  $\lambda$  represents the regularization parameter and  $i$  class.  
**Step-3:** Perform projection of the new test sample  $X$  onto the subspace of each class  $i$  using  $\hat{\alpha}$  as  $\hat{x}_i = X_i \hat{\alpha}_i$ .  
**Step-4:** Calculate distance between the test sample  $x$  and the class-specifics sub-sample  $\hat{x}_i$ .  
**Step-5:** Test sample  $x$  is assigned to that class whose distance is minimum.



### J. DATASET DESCRIPTION

The WSN dataset (WSN-DS) used in this study was published in [64]. According to this research, the WSN-DS was developed with the help of the Low Energy Adaptive Clustering Hierarchy (LEACH) routing protocol. SNs are used to collect the data and deliver it to the CH. The CH receives data from the SNs and transmits it to the BS. The BS then aggregates the data from all CHs and generates the dataset. This dataset has 18 features and five classes. The dataset's features are Node ID, Is CH, RSSI, Distance to CH, Average distance to CH, Max distance to CH, Current energy, ADV\_CH send, ADV\_CH receives, Distance CH to BS, Data send, Data received, etc. More details about the features are given in [64]. This dataset is divided into five different classes. The first class is normal, while the remaining classes are concerned with the DoS attacks. DoS attacks include Grayhole attacks, Blackhole attacks, Time division multiple access (TDMA) attacks, and Flooding attacks. The details about the classes and distribution of instances are given in Table 2. Furthermore, the WSN-DS consists of 374661 instances, with 340066 instances belong to the normal class, and 34595 instances belong to the malicious class. It indicates that WSN-DS is highly imbalanced, which could lead to a problem of weak generalization by classifiers.

TABLE 2. Details of WSN-DS.

Class	Label	Number of Instances
0	Normal	340066
1	Grayhole	14596
2	Blackhole	10049
3	TDMA	6638
4	Flooding	3312

### K. DATA SAMPLING

The WSN-DS is highly imbalanced, as mentioned above in Table 2. When data from the majority and minority classes is not balanced, it indicates a biasness in favour of the majority class. As a result, classification accuracy decreases, and the classifiers' performance degrades. In our case, the number of normal class instances is greater than the number of malicious class instances. Therefore, it is necessary to balance the data before giving it to the classification model. In literature, two types of balancing techniques are used to deal with imbalanced data. One is oversampling, and the second is undersampling. The oversampling increases the number of instances, whereas, the undersampling decreases the number of instances. Both are used to solve the problem of data imbalance. Both have their own sets of benefits and drawbacks. This research uses the Synthetic Minority Oversampling Technique (SMOTE) to handle the imbalanced data [66]. It duplicates the minority class instances by using an existing instance to make new instances.

## IV. RESULTS AND DISCUSSION

The proposed model's simulation results are the content of this section. In the proposed model, we adopted the WSN

model used in [64] with a slight modification. The modification is that in the proposed model, two BSs are used, while in [64], only one BS is used. The reason is that for implementing blockchain in our scenario, at least two BSs are required, while the model in [64] is a centralized model. Moreover, in this research, we use Google Collaboratory and the Python programming language to evaluate the performance of the proposed model using ML classifiers on both datasets: original WSN-DS and balanced WSN-DS. Furthermore, Solidity language is used to develop Smart contracts, implemented in Remix IDE. Also, the Remix web3 environment is integrated with the Ethereum wallet using Metamask. Then, virtual currency is transferred to the Ethereum wallet accounts using Ontology and Ganache to evaluate VBFT and PoW test networks' transaction costs. Furthermore, IPFS is installed on Windows, and Visual Studio code is used to upload and download files using IPFS. The hardware specifications are an Intel(R) Core(TM) i5-4200U processor running at 1.60GHz with 12GB of RAM and a 64-bit operating system.

### A. BLOCKCHAIN RESULTS' DISCUSSION

The results of the blockchain experiment are analyzed in this section. The proposed model is evaluated using two consensus mechanisms. Figure 3 compares the average transaction costs of the VBFT and PoW consensus mechanisms. There are three types of functions that are used in the proposed model: Reg(), Auth(), and Revoke(). WSN nodes are registered using the Reg() function, while the Auth() function confirms whether the node's identity exists or not. If the WSN nodes are validated, they are allowed to join the network. When the ML model discovers a malicious node in the network, the Revoke() function is called. We observe that the cost of a VBFT transaction is lower than the cost of a PoW transaction. This is because VBFT selects random verifiers in each round, reducing the probability of malicious nodes. In contrast, PoW selects mining nodes with a large number of processing resources in each round, leading to a high PoW transaction cost. Moreover, the transaction costs of Auth() and Revoke() in both PoW and VBFT are nearly the same. The reason is that both Auth() and Revoke() use a single attribute for each node. However, the cost of Reg() is different. The reason is that Reg() function uses three attributes for each node to complete the registration. PoW transaction is more costly than VBFT because it chooses the miner with the highest computational power to validate the transaction. In the case of VBFT, it selects the random verifiable for the miner who validates the transaction.

Figures 4(a) and 4(b) show how much time it takes to upload and download a file using IPFS. IPFS is used to upload and download five different files with sizes of 10kB, 100kB, 1MB, 10MB, and 100MB. The upload time of 10kB, 100kB, 1MB, 10MB, and 100MB files is 2.93s, 3.03s, 4.00s, 4.52s, and 5.01s, respectively. At the same time, the download time of the files is 2.22s, 2.44s, 3.25s, 3.7s, and 4.0s, respectively. The computing time increases as the size of files increases. We notice that the file's upload time is higher than its

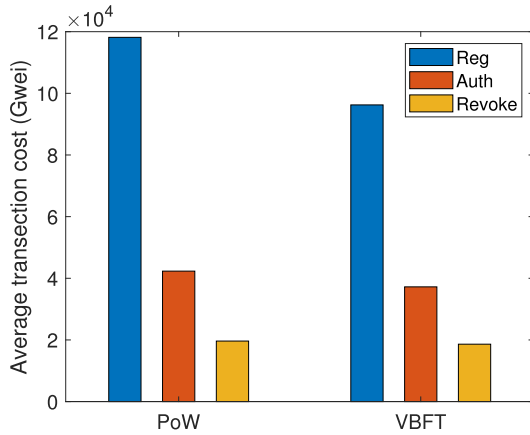


FIGURE 3. Comparison of PoW and VBFT consensus mechanism in terms of transaction cost.

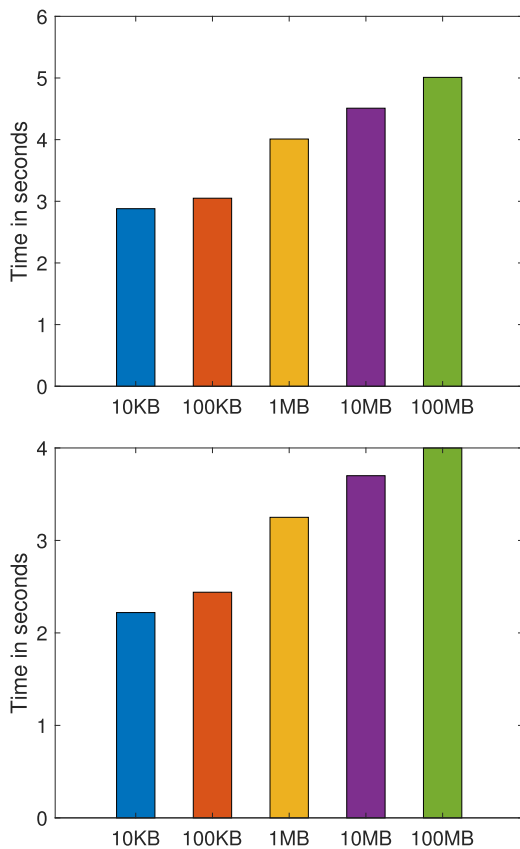


FIGURE 4. (a) Comparison of time consumed in uploading files on IPFS. (b) Comparison of time consumed in downloading files from IPFS.

download time. It also has the potential to be compared with the time required for normal file storage.

**B. ANALYSIS OF ML RESULTS**

The analysis of the simulation results is presented in this section. The proposed model is evaluated using six classifiers. Each classifier is trained on the original dataset and the balanced dataset balanced, and for a fair comparison,

all classifiers run on default settings. The proposed model uses several metrics such as accuracy, precision, recall, micro F1-score, and macro F1-score to evaluate the multi-label classification.

Figure 5(a) compares six ML classifiers using a balanced WSN-DS dataset. In the balanced dataset, the HGB classifier achieves an accuracy of 99%. As given in the figures, the accuracy of the GB, AdaBoost, LDA, XGB, and ridge classifiers is 98%, 93%, 88%, 97%, and 87%, respectively. The ridge classifier and the AdaBoost classifier give low accuracy because the ridge classifier first transforms the target variable into binary form and then treats it as a ridge regressor. Whereas, AdaBoost is the first ensemble learning boosting algorithm that uses multiple DTs with decision stumps and changes incorrect prediction weights. Furthermore, GB, XGB, and HGB are the modified variants of the AdaBoost technique. So they provide outstanding accuracy. On the original dataset, the HGB classifier obtains 99% accuracy. As demonstrated in Figure 5(b), the accuracy of GB, AdaBoost, LDA, XGB and ridge classifiers is 98.5%, 97%, 95%, 98%, and 95%, respectively. Both LDA and the ridge classifier provide an accuracy of 95%. Both these classifiers show biasness towards the normal class. The HGB classifier achieves 99% accuracy on the balanced and original datasets, which is more than all other classifiers. The reason is that HGB uses the binning method. The binning method sorts data features and distributes the sorted features evenly. After distributing the data in bins, it trains the weak learners, which makes the accuracies higher than those of other classifiers.

Precision is used to find the quality of a positive prediction provided by the model. Figure 6(a) shows the precision results of the classifiers on the balanced dataset. The precision value of GB, AdaBoost, LDA, XGB, HGB, and ridge classifiers is 98%, 94%, 89%, 98%, 99% and 89%, respectively. For the imbalanced dataset, the classifiers’ precision values are 97%, 90%, 81%, 97%, 98% and 83%, respectively as shown in Figure 6(b). The HGB classifier achieved 99% precision on the balanced dataset and 98% precision on the original dataset, which is greater than all other classifiers. Also, it indicates that the HGB classifier correctly classifies maximum positive samples. The reason is that the HGB classifier first sorts data features and then distributes the sorted features evenly. After that, it uses the weak learners for prediction, which makes higher precision as compared to other classifiers.

The recall is used to measure the positive predictions out of all positive predictions. Figure 7(a) depicts the balanced dataset’s recall values for different classifiers. In the balanced dataset, the recall for GB, AdaBoost, LDA, XGB, HGB, and ridge classifiers is 98%, 93%, 88%, 98%, 99%, and 87%, respectively. The recall value of the classifiers for the original dataset is 97%, 85%, 80%, 97%, 97%, and 71%, respectively, as shown in Figure 7(b). The HGB classifier achieved the highest recall of all classifiers, with values of 99% on the balanced dataset and 97% original dataset. It demonstrates that the HGB classifier accurately predicted the positive

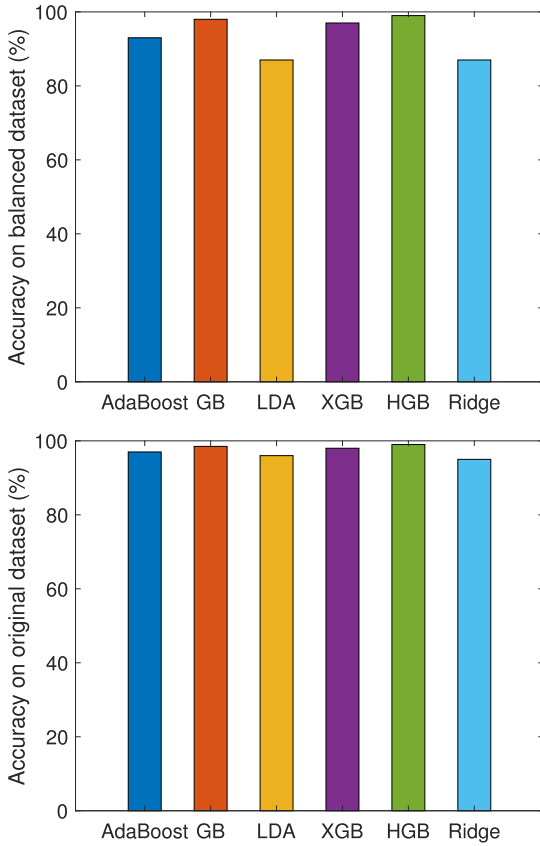


FIGURE 5. (a) Accuracy of classifiers on the balanced dataset. (b) Accuracy of classifiers on the original dataset.

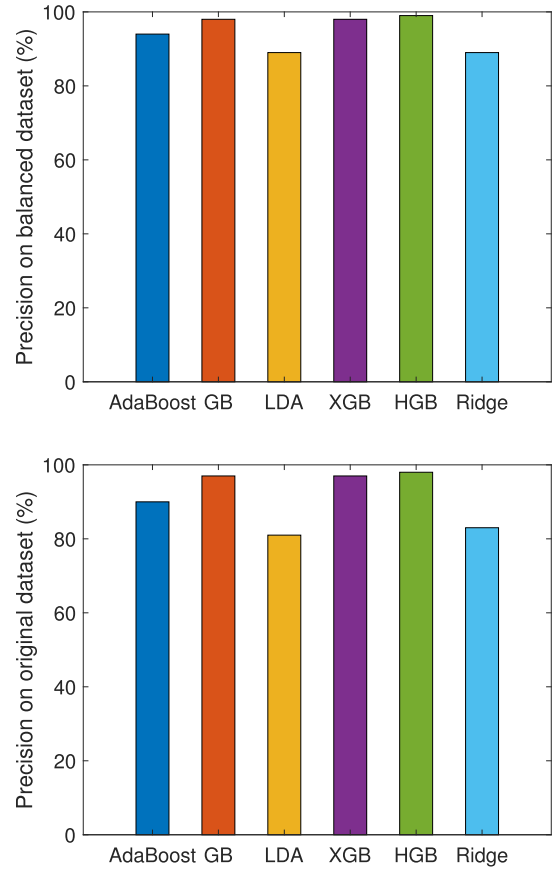


FIGURE 6. (a) Precision of classifiers on the balanced dataset. (b) Precision of classifiers on the original dataset.

samples out of all positive samples. The reason is that the HGB classifier organizes data features before equally distributing them. Then it employs the weak learners sequentially for prediction.

F1-score is the harmonic mean of precision and recall. It determines the single value that balances both precision and recall. The F1-scores of the classifiers differ for the balanced and original datasets, as shown in Figures 8(a) and 8(b). HGB classifier achieves the highest F1-score as compared to GB, AdaBoost, LDA, XGB, and ridge. HGB achieves a 99.5% F1-score on the balanced dataset, while other classifiers achieve 98%, 93%, 88%, 98%, and 87%, F1-score respectively. HGB achieves a 97% F1-score for the original dataset, while the other classifiers achieve 97%, 84%, 79%, 97%, and 75% F1-score, respectively. The HGB classifier obtains a high F1-score because precision and recall are both high on balanced and original datasets.

Figure 9 shows the receiver operating characteristic (ROC) curves of WSN-DS for multiclass using HGB classifier. The ROC area under the curve (AUC) is utilized for overall assessment. The ROC curves show all possible differences between true positive rate (TPR) and false positive rate (FPR) across multiple decision thresholds. The AUC evaluation metric converts this curve into a value between [0.5, 1]. A value of 1 indicates that the classifier performs efficiently, while

TABLE 3. Computational complexity of classifiers.

Classifiers	Training/Prediction Time (s)		Time Complexity
	Original Dataset	Balanced Dataset	
GB	317 sec	2880 sec	$O(ftn)$
AdaBoost	27 sec	109 sec	$O(ftn \log n)$
LDA	2 sec	9 sec	$O(Md^2)$
XGB	88 sec	700 sec	$O(dtx \log n)$
HGB	12 sec	81 sec	$O(ft(n\_bins))$
Ridge	1 sec	3 sec	$O(n^3)$

0.5 means it performs poorly. In Figure 9, the point closer to the top left corner, which is equal to 1, denotes higher classification results.

The time complexity of six classifiers in terms of training and prediction time is shown in Table 3. We execute six classifiers on the original and balanced datasets, and record the training and prediction times in seconds (s). Table 3 shows that the ridge and LDA classifiers achieve the best time complexity on the original and balanced datasets, while GB and XGB obtain the worst time complexity. The reason is that ridge and LDA are weak classifiers. Whereas, GB and XGB use many DTs, which perform better than LDA and ridge. The proposed HGB classifier takes more time than LDA and ridge, but it takes less time than boosting classifiers

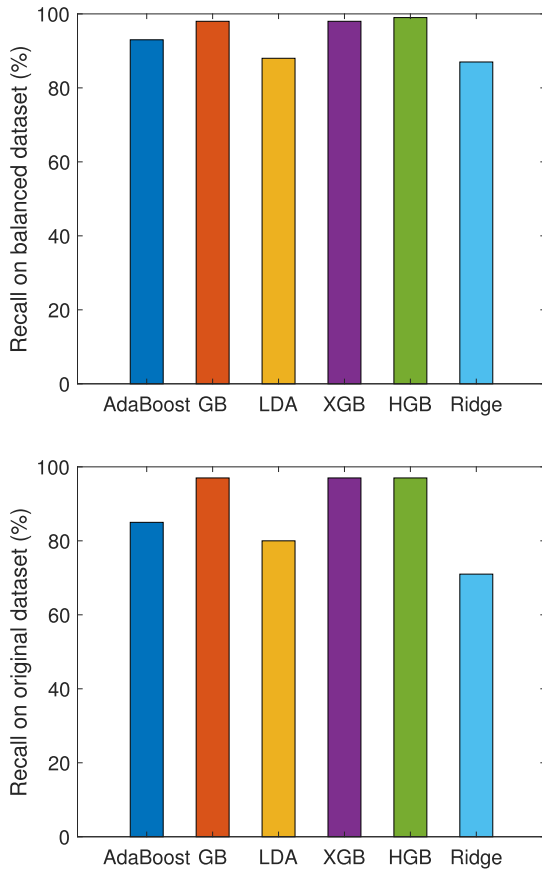


FIGURE 7. (a) Recall of classifiers on the balanced dataset. (b) Recall of classifiers on the original dataset.

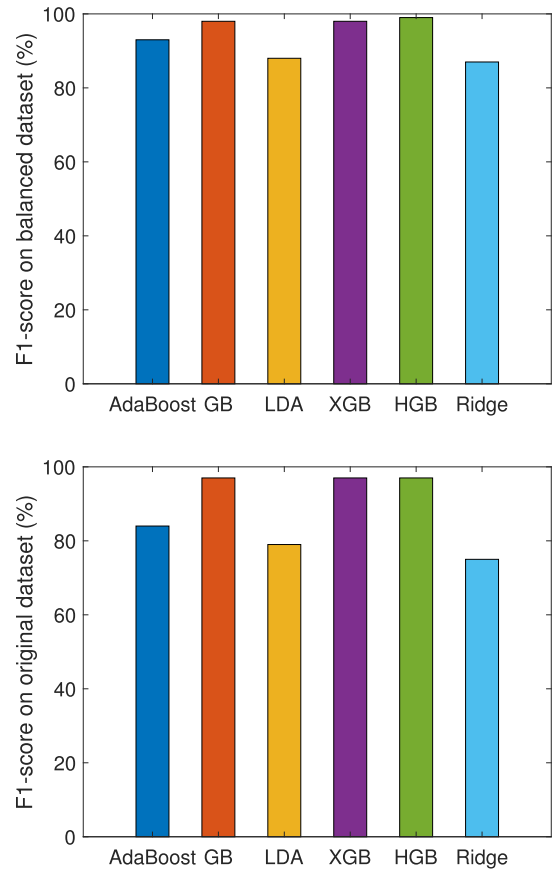


FIGURE 8. (a) F1-score of classifiers on the balanced dataset. (b) F1-score of classifiers on the original dataset.

and generates the best results. Furthermore, Table 3 shows the general time complexity of all ML classifiers.

The proposed model’s time complexity is given as the sum of the time complexity of each proposed technique. Firstly, blockchain is used to register and authenticate the sensor nodes. Its time complexity is  $O(n)$  because it uses sequential loops. Secondly, IPFS is used to store the data. Its time complexity is  $O(1)$ . Lastly, HGB is used to classify the normal and malicious nodes. Its time complexity is  $O(ft(n\_bins))$ . We compute the proposed model’s overall time complexity by adding all the above mentioned time complexities. The overall time complexity of the proposed model is given in equation (1) as follows.

$$T(t) = O(n) + O(1) + O(ft(n\_bins)) = O(ft(n\_bins)) \quad (1)$$

### C. COMPARATIVE ANALYSIS OF THE PROPOSED MODEL

This section evaluates the proposed model’s efficiency in comparison to state-of-the-art ML classifiers with respect to attacks, and micro and macro F1- scores. The proposed model is evaluated using six different classifiers: AdaBoost, GB, XGB, LDA, ridge, and HGB. SMOTE is used to balance the WSN-DS, and each classifier is trained on the original and balanced datasets. The performance metrics show that the HGB multi-label classifier gives the highest result in terms of micro and macro F1-score among the six classifiers. HGB performs more efficiently on the balanced dataset than on the

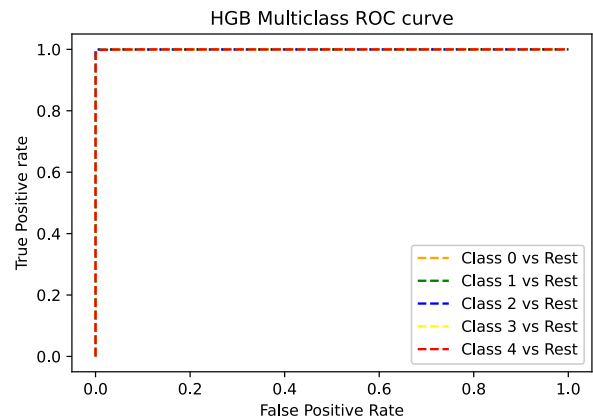


FIGURE 9. ROC curve of classifiers on the balanced dataset.

original dataset. The reason is that it is an ensemble learning classifier that uses multiple weak learners. Secondly, it uses the binning method, which distributes the data features into bins. After the data feature distribution in bins, HGB trains the weak learners on distributed data features, which gives them higher scores than others. Therefore, we use the HGB classifier combined with the SMOTE method in our model for malicious node detection.

We compute F1-scores for the detection performance of six classifiers using the original and balanced datasets. The results are presented in Tables 4 and 5. When computing

**TABLE 4.** The detection performance of six classifiers using the dataset balanced through SMOTE.

Classifiers	Black-hole	Flooding	Gray-hole	Normal	TDMA	Micro-F1	Macro-F1
GB	98	99	97	97	97	98	98
AdaBoost	87	99	87	96	96	93	93
LDA	82	99	71	96	91	88	88
XGB	98	100	98	97	97	98	98
HGB	100	100	100	99	99	99	99
Ridge	80	99	69	95	90	87	87

**TABLE 5.** The detection performance of six classifiers on the original dataset.

Classifiers	Black-hole	Flooding	Gray-hole	Normal	TDMA	Micro-F1	Macro-F1
GB	98	95	97	99	93	98	97
AdaBoost	78	93	56	99	94	97	84
LDA	73	85	58	99	79	95	79
XGB	98	95	95	99	96	99	97
HGB	99	95	98	100	95	99	97
Ridge	54	82	61	99	79	95	75

micro-F1, the average metric is calculated by averaging all classes' contributions. In contrast, in macro-F1, the metrics are computed separately for each class and then they are averaged. As a result, each class is treated equally, regardless of the number of samples in each class. Tables 4 and 5 show that HGB gives the highest micro F1-score and macro F1-score among all classifiers. Table 5 shows that both micro-F1 and macro-F1 values are equal in the balanced dataset because the number of test samples for each class is equal. However, in Table 5, the micro-F1 values of each class for the balanced dataset are higher than the macro-F1 values for the original dataset. The reason is that the number of test samples for each class is large, and the F1-score values are high.

In conclusion, we use numerous performance metrics for evaluating the proposed model. On the balanced and original datasets, the proposed model achieves high accuracy, precision, recall, and F1-score. There are many reasons for this enhanced performance. First, it is an ensemble learning boosting technique that uses many DTs consecutively to train the classifier. Second, it uses the binning method on the dataset, which makes HGB classifier different from other boosting techniques. The binning method organizes data features and uniformly distributes them. After organizing the data into bins, HGB uses several DTs to train the classifiers. This strategy of the HGB classifier gives better results than other boosting classifiers. In the proposed work, the simulations are performed using the WSN-DS. The proposed work is scalable and efficient enough to be deployed using the Empirical Dataset Generation Framework (EDGF) for WSNs [67].

## V. FEASIBILITY OF THE PROPOSED MODEL

In the proposed work, ML and blockchain are used in the WSN. The blockchain brings the benefit of tackling the security issues in WSNs. As WSNs comprise a large number of SNs, so, there exists an issue of efficiently distinguishing the malicious nodes from the legitimate nodes. For this purpose, ML techniques are employed. The techniques help in

classifying the nodes as malicious and legitimate. Once the model is trained, it will check for every incoming node and remove it if it is malicious before it becomes a part of the network. Furthermore, when the proposed model is applied on a similar network, it helps the network to perform in the presence of legitimate nodes only. The WSN is operated in an uncontrolled environment. With the application of ML, the nodes acting maliciously in the network are identified. The identified malicious nodes are removed from the network and only the legitimate nodes are used in routing activities. Thereby, making the system to operate in a controlled environment. Besides, the SNs do not have any heavy computational requirements. Regular SNs send their data to the CHs. CHs' storage capacity and computational power are higher than those of SNs. The CHs receive the data from the neighboring nodes and forward it to the BS. This task is conventional in the WSN and can be implemented easily in practice. The ML classification to detect the malicious nodes is performed at the BSs and not at the CHs. The BS has access to the electricity grid, has high computing power and can be equipped with the necessary hardware to accomplish the classification task. In other words, we should clarify that the SNs do not incur heavy computational tasks, and all parts of the network can be implemented easily in practice including blockchain, etc.

## VI. CONCLUSION

This study proposes a network model to detect malicious nodes in WSNs. SNs and CHs are registered on BSs that are responsible for monitoring the whole network and storing the credentials of the network nodes. In addition to this, blockchain technology is deployed on BSs. Both the verification and registration of nodes are done through blockchain. Moreover, a consensus mechanism, VBFT, is used to validate the transactions, which reduces transaction costs. Moreover, the network nodes' credentials and the hash values that IPFS produces are stored in the blockchain. Furthermore, the ML classifier, referred to as HGB, is utilized to identify malicious nodes. The simulation results show that the HGB classifier outperforms AdaBoost, GB, LDA, XGB, and ridge classifiers in terms of accuracy, precision, recall, micro-F1 score, and macro-F1 score.

In the proposed work, individual classifiers are used for classification, which does not provide enhanced efficiency. Moreover, the proposed work lacks in providing the vulnerability analysis of the smart contracts, which deteriorates the practicality of the work in the real world. Moreover, the monitoring of the WSN/IoT systems is beyond our scope at the current instant. In the future, to tackle the mentioned issues, a stacking model will be used in WSNs for performing more efficient malicious node detection. Furthermore, the Oyente tool will be used to assess the smart contracts' vulnerabilities. Moreover, this research will be conducted in various sectors using real-world networks. Besides, we aim to perform WSN/IoT system monitoring, as in [68], in the future.

## ACKNOWLEDGMENT

The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Education in Saudi Arabia for funding this research work through the project number (IF-PSAU-2021/01/19156).

## REFERENCES

- [1] O. J. Pandey, V. Gautam, S. Jha, M. K. Shukla, and R. M. Hegde, "Time synchronized node localization using optimal H-node allocation in a small world WSN," *IEEE Commun. Lett.*, vol. 24, no. 11, pp. 2579–2583, Nov. 2020, doi: [10.1109/LCOMM.2020.3008086](https://doi.org/10.1109/LCOMM.2020.3008086).
- [2] L. Xiong, N. Xiong, C. Wang, X. Yu, and M. Shuai, "An efficient lightweight authentication scheme with adaptive resilience of asynchronization attacks for wireless sensor networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 9, pp. 5626–5638, Sep. 2021, doi: [10.1109/TSMC.2019.2957175](https://doi.org/10.1109/TSMC.2019.2957175).
- [3] H. Wang, P. Tu, P. Wang, and J. Yang, "A redundant and energy-efficient clusterhead selection protocol for wireless sensor network," in *Proc. 2nd Int. Conf. Commun. Softw. Netw.*, 2010, pp. 554–558, doi: [10.1109/ICCSN.2010.46](https://doi.org/10.1109/ICCSN.2010.46).
- [4] S. A. Sert, E. Onur, and A. Yazici, "Security attacks and countermeasures in surveillance wireless sensor networks," in *Proc. 9th Int. Conf. Appl. Inf. Commun. Technol. (AICT)*, Oct. 2015, pp. 201–205.
- [5] S. A. Sert, C. Fung, R. George, and A. Yazici, "An efficient fuzzy path selection approach to mitigate selective forwarding attacks in wireless sensor networks," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6.
- [6] R. Alkhudary, "Blockchain technology between Nakamoto and supply chain management: Insights from academia and practice," *SSRN Electron. J.*, pp. 1–12, Jul. 2020, doi: [10.2139/ssrn.3660342](https://doi.org/10.2139/ssrn.3660342).
- [7] Z. Abubaker, N. Javaid, A. Almogren, M. Akbar, M. Zuair, and J. Ben-Othman, "Blockchain service provisioning and malicious node detection via federated learning in scalable internet of sensor things networks," *Comput. Netw.*, vol. 204, Feb. 2022, Art. no. 108691.
- [8] A. S. Yahaya, N. Javaid, M. U. Javed, A. Almogren, and A. Radwan, "Blockchain based secure energy trading with mutual verifiable fairness in a smart community," *IEEE Trans. Ind. Informat.*, vol. 18, no. 11, pp. 7412–7422, Nov. 2022.
- [9] Z. Abubaker, A. U. Khan, A. Almogren, S. Abbas, A. Javaid, A. Radwan, and N. Javaid, "Trustful data trading through monetizing IoT data using Blockchain based review system," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 5, p. e6739, Feb. 2022.
- [10] S. Abbas, N. Javaid, A. Almogren, S. M. Gulfam, A. Ahmed, and A. Radwan, "Securing genetic algorithm enabled SDN routing for blockchain based Internet of Things," *IEEE Access*, vol. 9, pp. 139739–139754, 2021.
- [11] Z. Li, A. V. Barenji, and G. Q. Huang, "Toward a blockchain cloud manufacturing system as a peer to peer distributed network platform," *Robot. Comput.-Integr. Manuf.*, vol. 54, pp. 133–144, Dec. 2018.
- [12] A. Wellington dos Santos Abreu, E. F. Coutinho, and C. Ilane Moreira Bezerra, "Performance evaluation of data transactions in blockchain," *IEEE Latin Amer. Trans.*, vol. 20, no. 3, pp. 409–416, Mar. 2022, doi: [10.1109/TLA.2022.9667139](https://doi.org/10.1109/TLA.2022.9667139).
- [13] G. Kumar, R. Saha, M. Rai, R. Thomas, and T. H. Kim, "Proof-of-work consensus approach in blockchain technology for cloud and fog computing using maximization-factorization statistics," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6835–6842, Aug. 2019, doi: [10.1109/JIOT.2019.2911969](https://doi.org/10.1109/JIOT.2019.2911969).
- [14] J. Benet, "IPFS—content addressed, versioned, P2P file system," 2014, *arXiv:1407.3561*.
- [15] G. Kolumban-Antal, V. Lasak, R. Bogdan, and B. Groza, "A secure and portable multi-sensor module for distributed air pollution monitoring," *Sensors*, vol. 20, no. 2, p. 403, Jan. 2020, doi: [10.3390/s20020403](https://doi.org/10.3390/s20020403).
- [16] Y. Xu, J. Ren, G. Wang, C. Zhang, J. Yang, and Y. Zhang, "A blockchain-based nonrepudiation network computing service scheme for industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3632–3641, Jun. 2019, doi: [10.1109/TII.2019.2897133](https://doi.org/10.1109/TII.2019.2897133).
- [17] B. Jia, T. Zhou, W. Li, Z. Liu, and J. Zhang, "A blockchain-based location privacy protection incentive mechanism in crowd sensing networks," *Sensors*, vol. 18, no. 11, p. 3894, 2018, doi: [10.3390/s18113894](https://doi.org/10.3390/s18113894).
- [18] D. P. Kumar, A. Tarachand, and C. S. R. Annavarapu, "Machine learning algorithms for wireless sensor networks: A survey," *Inf. Fusion*, vol. 49, pp. 1–25, Sep. 2019.
- [19] S. Brotsis, K. Limniotis, G. Bendiab, N. Kolokotronis, and S. Shialeas, "On the suitability of blockchain platforms for IoT applications: Architectures, security, privacy, and performance," *Comput. Netw.*, vol. 191, May 2021, Art. no. 108005.
- [20] Z. Cui, F. Xue, S. Zhang, X. Cai, Y. Cao, W. Zhang, and J. Chen, "A hybrid blockchain-based identity authentication scheme for multi-WSN," *IEEE Trans. Services Comput.*, vol. 13, no. 2, pp. 241–251, Apr. 2020, doi: [10.1109/TSC.2020.2964537](https://doi.org/10.1109/TSC.2020.2964537).
- [21] J. Yang, S. He, Y. Xu, L. Chen, and J. Ren, "A trusted routing scheme using blockchain and reinforcement learning for wireless sensor networks," *Sensors*, vol. 19, no. 4, p. 970, Feb. 2019, doi: [10.3390/s19040970](https://doi.org/10.3390/s19040970).
- [22] Y. Tian, Z. Wang, J. Xiong, and J. Ma, "A blockchain-based secure key management scheme with trustworthiness in DWSNs," *IEEE Trans. Ind. Informat.*, vol. 16, no. 9, pp. 6193–6202, Sep. 2020, doi: [10.1109/TII.2020.2965975](https://doi.org/10.1109/TII.2020.2965975).
- [23] A. Moinet, B. Darties, and J. Baril, "Blockchain based trust & authentication for decentralized sensor networks," 2017, *arXiv:1706.01730*.
- [24] B. K. Mohanta, D. Jena, S. Ramasubbarreddy, M. Daneshmand, and A. H. Gandomi, "Addressing security and privacy issues of IoT using blockchain technology," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 881–888, Jan. 2021.
- [25] S. Mori, "Secure caching scheme by using blockchain for information-centric network-based wireless sensor networks," *J. Signal Process.*, vol. 22, no. 3, pp. 97–108, May 2018, doi: [10.2299/jsp.22.97](https://doi.org/10.2299/jsp.22.97).
- [26] W. She, Q. Liu, Z. Tian, J.-S. Chen, B. Wang, and W. Liu, "Blockchain trust model for malicious node detection in wireless sensor networks," *IEEE Access*, vol. 7, pp. 38947–38956, 2019, doi: [10.1109/ACCESS.2019.2902811](https://doi.org/10.1109/ACCESS.2019.2902811).
- [27] Y. Chang, H. Tang, Y. Cheng, Q. Zhao, and B. Yuan, "Dynamic hierarchical energy-efficient method based on combinatorial optimization for wireless sensor networks," *Sensors*, vol. 17, no. 7, p. 1665, Jul. 2017.
- [28] Y. Chang, X. Yuan, B. Li, D. Niyato, and N. Al-Dhahir, "A joint unsupervised learning and genetic algorithm approach for topology control in energy-efficient ultra-dense wireless sensor networks," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2370–2373, Nov. 2018.
- [29] T.-H. Kim, R. Goyat, M. K. Rai, G. Kumar, W. J. Buchanan, R. Saha, and R. Thomas, "A novel trust evaluation process for secure localization using a decentralized blockchain in wireless sensor networks," *IEEE Access*, vol. 7, pp. 184133–184144, 2019, doi: [10.1109/ACCESS.2019.2960609](https://doi.org/10.1109/ACCESS.2019.2960609).
- [30] S. Rathore, B. W. Kwon, and J. H. Park, "BlockSecIoTNet: Blockchain-based decentralized security architecture for IoT network," *J. Netw. Comput. Appl.*, vol. 143, pp. 167–177, Oct. 2019, doi: [10.1016/j.jnca.2019.06.019](https://doi.org/10.1016/j.jnca.2019.06.019).
- [31] P. Kumar, G. P. Gupta, and R. Tripathi, "TP2SF: A trustworthy privacy-preserving secured framework for sustainable smart cities by leveraging blockchain and machine learning," *J. Syst. Archit.*, vol. 115, May 2021, Art. no. 101954.
- [32] P. Kumar, R. Kumar, G. Srivastava, G. P. Gupta, R. Tripathi, T. R. Gadekallu, and N. N. Xiong, "PPSF: A privacy-preserving and secure framework using blockchain-based machine-learning for IoT-driven smart cities," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2326–2341, Jul. 2021.
- [33] Z. Li, H. Guo, W. M. Wang, Y. Guan, A. V. Barenji, G. Q. Huang, K. S. McFall, and X. Chen, "A blockchain and AutoML approach for open and automated customer service," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3642–3651, Jun. 2019.
- [34] S. Seth, K. K. Chahal, and G. Singh, "A novel ensemble framework for an intelligent intrusion detection system," *IEEE Access*, vol. 9, pp. 138451–138467, 2021, doi: [10.1109/ACCESS.2021.3116219](https://doi.org/10.1109/ACCESS.2021.3116219).
- [35] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "A lightweight blockchain based framework for underwater IoT," *Electronics*, vol. 8, no. 12, p. 1552, Dec. 2019, doi: [10.3390/electronics8121552](https://doi.org/10.3390/electronics8121552).
- [36] Y. Ren, Y. Liu, S. Ji, A. K. Sangaiah, and J. Wang, "Incentive mechanism of data storage based on blockchain for wireless sensor networks," *Mobile Inf. Syst.*, vol. 2018, pp. 1–10, Aug. 2018, doi: [10.1155/2018/6874158](https://doi.org/10.1155/2018/6874158).
- [37] G. Rathee, M. B. K. Prabhu, C. Sharmi, and D. Gupta, "A secure IoT sensors communication in industry 4.0 using blockchain technology," *J. Ambient Intell. Humanized Comput.*, vol. 12, pp. 533–545, Apr. 2020, doi: [10.1007/s12652-020-02017-8](https://doi.org/10.1007/s12652-020-02017-8).

- [38] Y. Liu, K. Wang, S. Member, Y. Lin, W. Xu, and S. Member, "LightChain: A lightweight blockchain system for industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3571–3581, Jun. 2019, doi: [10.1109/TII.2019.2904049](https://doi.org/10.1109/TII.2019.2904049).
- [39] P. Danzi, S. Member, A. E. Kalør, and S. Member, "Delay and communication tradeoffs for blockchain systems with lightweight IoT clients," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2354–2365, Apr. 2019, doi: [10.1109/JIOT.2019.2906615](https://doi.org/10.1109/JIOT.2019.2906615).
- [40] A. Rovira-Sugranes and A. Razi, "Optimizing the age of information for blockchain technology with applications to IoT sensors," *IEEE Commun. Lett.*, vol. 24, no. 1, pp. 183–187, Jan. 2020, doi: [10.1109/LCOMM.2019.2949557](https://doi.org/10.1109/LCOMM.2019.2949557).
- [41] X. Ding, J. Guo, D. Li, and W. Wu, "An incentive mechanism for building a secure blockchain-based Internet of Things," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 1, pp. 477–487, Jan. 2021.
- [42] J. Huang, L. Kong, G. Chen, M.-Y. Wu, X. Liu, and P. Zeng, "Towards secure industrial IoT: Blockchain system with credit-based consensus mechanism," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3680–3689, Jun. 2019.
- [43] P. K. Sharma and J. H. Park, "Blockchain based hybrid network architecture for the smart city," *Future Gener. Comput. Syst.* vol. 86, pp. 650–655, Sep. 2018, doi: [10.1016/j.future.2018.04.060](https://doi.org/10.1016/j.future.2018.04.060).
- [44] K. Haseeb, N. Islam, A. Almgren, and I. U. Din, "Intrusion prevention framework for secure routing in WSN-based mobile Internet of Things," *IEEE Access*, vol. 7, pp. 185496–185505, 2019, doi: [10.1109/ACCESS.2019.2960633](https://doi.org/10.1109/ACCESS.2019.2960633).
- [45] S. Kushch and F. Prieto-Castrillo, "A rolling blockchain for a dynamic WSNs in a smart city," 2018, *arXiv:1806.11399*.
- [46] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11008–11021, Nov. 2018, doi: [10.1109/TVT.2018.2866365](https://doi.org/10.1109/TVT.2018.2866365).
- [47] M. H. Kumar, V. M. Y. Suresh, and J. S. G. Nagalalli, "RETRACTED ARTICLE: Trust aware localized routing and class based dynamic block chain encryption scheme for improved security in WSN," *J. Ambient Intell. Humanized Comput.*, vol. 12, pp. 5287–5295, Apr. 2020, doi: [10.1007/s12652-020-02007-w](https://doi.org/10.1007/s12652-020-02007-w).
- [48] R. Goyat, G. Kumar, M. K. Rai, R. Saha, R. Thomas, and T. H. Kim, "Blockchain powered secure range-free localization in wireless sensor networks," *Arabian J. Sci. Eng.*, vol. 45, no. 8, pp. 6139–6155, Aug. 2020, doi: [10.1007/s13369-020-04493-8](https://doi.org/10.1007/s13369-020-04493-8).
- [49] Y. Chang, X. Yuan, B. Li, D. Niyato, and N. Al-Dhahir, "Machine-learning-based parallel genetic algorithms for multi-objective optimization in ultra-reliable low-latency WSNs," *IEEE Access*, vol. 7, pp. 4913–4926, 2019.
- [50] B. Schölkopf, Z. Luo, and V. Vovk, *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik*. New York, NY, USA: Springer, 2013, pp. 1–287, doi: [10.1007/978-3-642-41136-6](https://doi.org/10.1007/978-3-642-41136-6).
- [51] S. Chaudhury. *Tuning of Adaboost With Computational Complexity*. Medium.com. Accessed: Jan. 17, 2023. [Online]. Available: <https://medium.com/@chaudhuryrijani/tuning-of-adaboost-with-computational-complexity-8727d01a9d20>
- [52] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers Neurobot.*, vol. 7, p. 21, Dec. 2013, doi: [10.3389/fnbot.2013.00021](https://doi.org/10.3389/fnbot.2013.00021).
- [53] Prashant. *Computational Complexity of Machine Learning Algorithm*. Medium.com. Accessed: Jan. 17, 2023. [Online]. Available: <https://medium.com/analytics-vidhya/computational-complexity-of-ml-algorithms-1bdc88af1c7a>
- [54] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," *AI Commun.*, vol. 30, no. 2, pp. 169–190, 2017, doi: [10.3233/AIC-170729](https://doi.org/10.3233/AIC-170729).
- [55] M. Krause. *Computational Complexity of Linear Discrimination Analysis*. Stackexchange.com. Accessed: Jan. 17, 2023. [Online]. Available: <https://stats.stackexchange.com/questions/142565/computational-complexity-for-linear-discriminant-analysis>
- [56] D.-K. Choi, "Data-driven materials modeling with XGBoost algorithm and statistical inference analysis for prediction of fatigue strength of steels," *Int. J. Precis. Eng. Manuf.*, vol. 20, no. 1, pp. 129–138, Jan. 2019, doi: [10.1007/s12541-019-00048-6](https://doi.org/10.1007/s12541-019-00048-6).
- [57] M. Virgolin. *Time Complexity for Different Machine Learning Algorithm*. Marcovirgolin.github.io. Accessed: Jan. 17, 2023. [Online]. Available: [https://marcovirgolin.github.io/extras/details\\_time\\_complexity\\_machine\\_learning\\_algorithms](https://marcovirgolin.github.io/extras/details_time_complexity_machine_learning_algorithms)
- [58] S. Premanand. *Histogram Boosting Gradient Classifier*. Analyticsvidhya.com. Accessed: Jan. 17, 2023. [Online]. Available: <https://www.analyticsvidhya.com/blog/2022/01/histogram-boosting-gradient-classifier>
- [59] A. Guryanov, "Histogram-based algorithm for building gradient boosting ensembles of piecewise linear decision trees," in *Proc. Int. Conf. Anal. Images, Social Netw. Texts*. Cham, Switzerland: Springer, Jul. 2019, pp. 39–50, doi: [10.1007/978-3-030-37334-4](https://doi.org/10.1007/978-3-030-37334-4).
- [60] J. Garcia. *Histogram for Efficient Gradient Boosting*. Robotenique.github.io. Accessed: Jan. 17, 2023. [Online]. Available: <https://robotenique.github.io/posts/gbm-histogram>
- [61] B. Behera and G. Kumaravelan, "Performance evaluation of machine learning algorithms in biomedical document classification," *Performance Evaluation*, vol. 29, no. 5, pp. 5704–5716, 2020, doi: [10.1002/9781118150238.ch8](https://doi.org/10.1002/9781118150238.ch8).
- [62] G. C. McDonald, *Ridge Regression*. 2011, pp. 173–188, doi: [10.1002/9781118150238.ch8](https://doi.org/10.1002/9781118150238.ch8).
- [63] R. Yin, Y. Liu, W. Wang, and D. Meng, "Sketch kernel ridge regression using circulant matrix: Algorithm and theory," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3512–3524, Sep. 2020, doi: [10.1109/TNNLS.2019.2944959](https://doi.org/10.1109/TNNLS.2019.2944959).
- [64] I. Almomani, B. Al-Kasasbeh, and M. Al-Akhras, "WSN-DS: A dataset for intrusion detection systems in wireless sensor networks," *J. Sensors*, vol. 2016, pp. 1–16, 2016, doi: [10.1155/2016/4731953](https://doi.org/10.1155/2016/4731953).
- [65] C. Dehury, S. N. Srirama, P. K. Donta, and S. Dustdar, "Securing clustered edge intelligence with blockchain," *IEEE Consum. Electron. Mag.*, early access, Apr. 4, 2022, doi: [10.1109/MCE.2022.3164529](https://doi.org/10.1109/MCE.2022.3164529).
- [66] D. Rosadi, D. Arisanty, W. Andriyani, S. Peiris, D. Agustina, D. Dowe, and Z. Fang, "Improving machine learning prediction of peatlands fire occurrence for unbalanced data using SMOTE approach," in *Proc. Int. Conf. Data Sci., Artif. Intell., Bus. Anal. (DATABIA)*, Nov. 2021, pp. 160–163.
- [67] D. K. Sah, K. Cengiz, P. K. Donta, V. N. Inikollu, and T. Amgoth, "EDGF: Empirical dataset generation framework for wireless sensor networks," *Comput. Commun. J.*, vol. 180, pp. 48–56, Dec. 2021.
- [68] P. K. Donta and S. Dustdar, "The promising role of representation learning for distributed computing continuum systems," in *Proc. IEEE Int. Conf. Service-Oriented Syst. Eng. (SOSE)*, Aug. 2022, pp. 126–132.



**MUHAMMAD NOUMAN** received the bachelor's degree in computer science from the Federal Urdu University of Arts, Science and Technology (FUUAST), Islamabad, Pakistan, in 2019. He is currently pursuing the master's degree in computer science with the Communication Over Sensors (ComSens) Research Laboratory, Department of Computer Science, COMSATS University Islamabad, Pakistan, under the supervision of Dr. Nadeem Javaid. His research interests include

computer networks, wireless sensor networks, blockchain, and machine learning.



**UMAR QASIM** received the B.S. degree in computer science and the M.B.A. degree from Hamdard University and the M.S. and Ph.D. degrees in information systems from the New Jersey Institute of Technology, USA. He has been in the IT field for over 20 years and has taught at various post-secondary institutions in North America and Pakistan. He has extensive experience in the field of information technology. He worked at Dalhousie University, McMaster University, and various other software development companies in USA, for more than ten years.

He headed the Digital Preservation Program with the University of Alberta for eight years and was responsible for internal operations and external preservation partnerships. He is currently working as an Associate Professor with the University of Engineering and Technology (UET) Lahore, where he is involved in teaching and research and serving on various committees both at the department and university level. He maintained and shared expertise on digital preservation with the university community and the professional community of practice at large. His research interests include the various areas of information systems, including but not limited to wireless sensor networks, databases, digital preservation, data science, and information management.



**HINA NASIR** received the bachelor's degree in information and communication systems engineering from the NUST School of Electrical Engineering and Computer Science (SEECS), in 2008, the M.S. degree in computer science from International Islamic University (IIUI), in 2015, under the supervision of Dr. Nadeem Javaid, and the Ph.D. degree in computer science under the supervision of Dr. Nadeem Javaid. She worked as an Assistant Professor with the Department of Computer Science, Air University, Islamabad. Currently, she is doing postdoctoral research at Leeds University, U.K. Her research interests include wireless sensor networks, underwater wireless sensor networks, cooperative communication, cooperative routing, buffer-aided cooperative communication, energy harvesting in wireless networks, 5G networks, and the Internet of Things.

include wireless networks, cognitive radio networks, the Internet of Things, UAV-assisted networking, and RF energy harvesting. He was a recipient of the Best Paper Award at the IEEE Global Communications Conference 2018 on Ad Hoc and Sensors Networks Symposium.



**ABDULLAH ALMASOUD** (Member, IEEE) received the B.Sc. degree in computer engineering from King Saud University, Riyadh, Saudi Arabia, and the M.Sc. degree in computer engineering and the Ph.D. degree in computer engineering and electrical engineering from Iowa State University, Ames, IA, USA. Currently, he is an Assistant Professor with the Department of Electrical Engineering, Prince Sattam Bin Abdulaziz University, Al-Kharj, Saudi Arabia. His research interests

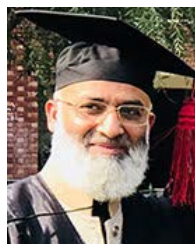
include wireless networks, cognitive radio networks, the Internet of Things, UAV-assisted networking, and RF energy harvesting. He was a recipient of the Best Paper Award at the IEEE Global Communications Conference 2018 on Ad Hoc and Sensors Networks Symposium.



**MUHAMMAD IMRAN** (Member, IEEE) worked with King Saud University (KSU), Saudi Arabia, as an Associate Professor. He is currently working as a Senior Lecturer with the School of Science, Engineering and Information Technology, Federation University, Australia. He is also the Founding Leader of the Wireless Networks and Security (WINS) Research Group, KSU, from 2013 to 2021. His research interests include mobile and wireless networks, the Internet of

Things, big data analytics, cloud/edge computing, and information security. His research is financially supported by several national and international grants. He has completed a number of international collaborative research projects with reputable universities. He has published more than 300 research articles in peer-reviewed, highly-reputable international conferences (90), journals (198), editorials (15), book chapter (1), and two edited books. Many of his research articles are among the highly cited and most downloaded. His research has been cited more than 11,500 with H-index of 55 and i-10 index of 175 (Google Scholar). He has received a number of awards and fellowships.

He served as the Editor-in-Chief for *European Alliance for Innovation (EAI) Transactions on Pervasive Health and Technology* and an Associate Editor for *IEEE Communications Magazine*. He is serving as an Associate Editor for top ranked international journals, such as *IEEE Network*, *Future Generation Computer Systems*, and *IEEE ACCESS*. He served/serving as a Guest Editor for about two dozen special issues in journals, such as *IEEE Communications Magazine*, *IEEE Wireless Communications Magazine*, *Future Generation Computer Systems*, *IEEE ACCESS*, and *Computer Networks*. He has been involved in about 100 peer-reviewed international conferences and workshops in various capacities, such as the chair, the co-chair, and a technical program committee member. He has been consecutively awarded with an Outstanding Associate Editor of *IEEE Access*, in 2018 and 2019, besides many others.



**NADEEM JAVAID** (Senior Member, IEEE) received the bachelor's degree in computer science from Gomal University, Dera Ismail Khan, Pakistan, in 1995, the master's degree in electronics from Quaid-i-Azam University, Islamabad, Pakistan, in 1999, and the Ph.D. degree from the University of Paris-Est, France, in 2010. He is currently a Tenured Professor and the Founding Director of the Communications Over Sensors (ComSens) Research Laboratory, Department of

Computer Science, COMSATS University Islamabad, Islamabad Campus. He has supervised 158 master's and 30 Ph.D. theses. He has authored over 900 articles in technical journals and international conferences. His research interests include energy optimization in smart/microgrids and in wireless sensor networks using data analytics and blockchain. He was a recipient of the Best University Teacher Award (BUTA'16) from the Higher Education Commission (HEC) of Pakistan, in 2016, and the Research Productivity Award (RPA'17) from the Pakistan Council for Science and Technology (PCST), in 2017.

...