Received 2 June 2022, accepted 19 August 2022, date of publication 29 August 2022, date of current version 9 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3202807

RESEARCH ARTICLE

STLGBM-DDS: An Efficient Data Balanced DoS Detection System for Wireless Sensor Networks on Big Data Environment

MURAT DENER[1], SAMED AL[1], AND ABDULLAH ORMAN[2]
[1]Department of Information Security Engineering, Graduate School of Natural and Applied Sciences, Gazi University, 06560 Ankara, Turkey
[2]Department of Computer Technologies, Yıldırım Beyazıt University, 06010 Ankara, Turkey

Corresponding author: Murat Dener (muratdener@gazi.edu.tr)

ABSTRACT Wireless Sensor Networks(WSNs) are vulnerable to a variety of unique security risks and threats in their data collection and transmission processes. One of the most common attacks on WSNs that can target all layers of the protocol stack is the DoS attack. In this study, a unique DoS Intrusion Detection System (DDS) is proposed to detect DoS attacks specific to WSNs. The proposed system is an ensemble intrusion detection system called STLGBM-DDS, which is developed on Apache Spark big data platform in Google Colab environment, combining LightGBM machine learning algorithm, data balancing and feature selection processes. In order to reduce the effects of data imbalance on system performance, data imbalance processing consisting of Synthetic Minority Oversampling Technique (SMOTE) and Tomek-Links sampling methods called STL was used. In addition, Information Gain Ratio was used as a feature selection technique in the data preprocessing stage. The effects of both data balancing and feature selection stages on the detection performance of the system were investigated. The results obtained were evaluated using the Accuracy, F-Measure, Precision, Recall, ROC Curve and Precision-Recall Curve parameters. As a result, the proposed method achieved an overall accuracy of 99.95%. Also, it achieved 99.99%, 99.96%, 99.98%, 99.92%, and 99.87% accuracy performance according to Normal, Grayhole, Blackhole, TDMA and Flooding classes, respectively. According to the results obtained, the proposed method has achieved very successful results in DoS attack detection in WSNs compared to current methods.

INDEX TERMS Wireless sensor networks, DoS attacks, intrusion detection, deep learning, imbalanced data.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) can be set up to monitor and collect data on physical or environmental conditions such as temperature, pressure, humidity, motion and sound. WSNs are an infrastructure of thousands of low-cost, limited-power, and multi-functional distributed sensor nodes that wirelessly interconnect to form an interoperable sensor domain. WSNs are a network of sensor nodes that can be part of the Internet of Things (IoT). The collected data is then processed, analyzed and presented to the user via base stations. WSNs have at least one base station that acts as a gateway between the sensor network and the outside world. Sensor nodes detect physical data and send the detected data to the base station via single-hop or multi-hop communication. WSNs represent a special class of ad hoc networks [1]. In principle, these network nodes have a mode of self-organization, as they are intended to be located quickly and dispersedly in an area of interest.

The WSN market is expected to grow significantly in the coming years due to the need for network infrastructures, developments in artificial intelligence, machine learning and big data. Recent developments in the Internet of Things and Artificial Intelligence have further increased the demand for wireless networks and seamless connections. The growth of the industrial wireless sensor network market is expected to increase as these technologies are rapidly adopted by the oil, gas, manufacturing, utilities and automotive industries [2].

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

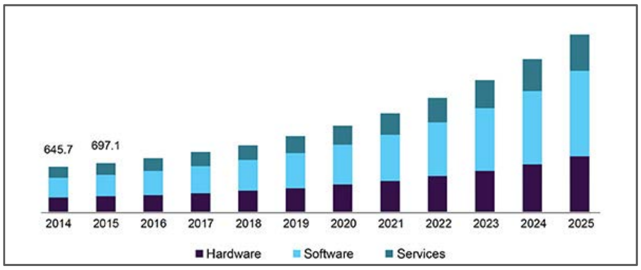


FIGURE 1. **U.S. Industrial WSN market size (USD Million)[2].**

In Fig. 1, the global wireless sensor network market size is given. The market size was valued at $3.28 billion in 2018 and is expected to grow by 15.2% from 2019 to 2025, reaching $8.67 billion by 2025 [2].

Today, research on WSNs has attracted a great deal of attention due to its wide variety of real-time applications such as critical military surveillance, battlefields, building security monitoring, farmland and forest monitoring, robotics, and healthcare. WSNs offer economical, flexible, scalable and pragmatic solutions in many situations. In order for these applications to work successfully and efficiently, all nodes must work collaboratively and reliably. Securing WSNs from attack is a difficult task for several reasons unique to these networks. In WSNs, the sensor nodes are densely located in an area known as the sensor area. These nodes have limited computing power and bandwidth and are managed remotely. As nodes often remain unattended within the WSNs, an adversary can easily capture a node. Also, the sensor nodes are prone to various failures and the communication medium is also unreliable. Therefore, security for WSNs is both a difficult and important task.

Many WSN applications require high availability. Therefore, it is important to deal with Denial of Service (DoS) attacks. Although work on detecting DoS attacks has become popular in recent years, it still remains a major challenge for WSNs today. While the use of DoS attack mitigation and detection techniques for traditional networks and systems is frequently investigated in the literature, effective detection methods of these attacks in WSNs need to be better understood and emphasized. DoS attacks on WSNs tend to have major effects, especially due to the constrained sensor devices that create them [3]. DoS attacks can be detected by tools known as Intrusion Detection Systems (IDS). IDSs monitor system behavior to detect and prevent malicious traffic. In this way, attacks can be easily detected by determining the normal traffic pattern and size in the network. IDS observes and analyzes events generated in the network to detect anything out of the ordinary and alert sensor nodes about an intruder [4], [5]. Using data collected from sensors, cyber threat analysts and intrusion detection/prevention systems can discover useful information in real time. This information can help detect vulnerabilities and attacks and develop security solutions accordingly.

In this study, a classification-based intrusion detection system specific to WSNs was implemented by using an ensemble method that combines LightGBM machine learning algorithm, data balancing and feature selection on Apache Spark big data platform in the Google Colab environment. WSN-DS dataset was used in the study. Since the WSN-DS dataset is imbalanced, it is combined with the LightGBM machine learning method and STL(SMOTE + Tomek-Link) data imbalance processing. In addition, the Information Gain Ratio feature selection technique is used to both increases detection performance and reduce processing load. Apache Spark environment is preferred because both speeds is important in attack detection and the data used is large. The studies were carried out using PySpark, which provides Python support. In the study, classes labeled as Normal, TDMA, Grayhole, Blockhole and Flooding in the WSN-DS dataset containing network flow data were classified as Random Forest, Decision Tree, Naive Bayes, Logistic Regression, Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), GRU(Gated Recurrent Units), CNN-LSTM and proposed method. The classification success of the proposed algorithm on the dataset was compared by using the evaluation parameters Accuracy, F-Measure, Precision, Recall, ROC Curves, and Precision-Recall Curve.

The main contributions of this study can be summarized as follows:

1) In the study, a classification-based DoS intrusion detection system specific to WSNs was developed and it was verified that it works effectively in the big data environment.

2) Another contribution of the study is that deep learning approaches have been verified to be more successful in intrusion detection systems than traditional machine learning methods.

3) LightGBM machine learning technique has been shown to be more successful than the hybrid deep learning approaches that have been popular in recent years in detecting WSNs-specific intrusions.

4) Feature selection was performed on the WSN-DS dataset in order to both reduce the computational complexity and increase the classification accuracy. As a result of this process, more meaningful features were used for attack detection. In addition, a faster IDS has been developed since fewer data will be processed. The performance improvement is confirmed by the results obtained.

5) SMOTE oversampling and Tomek-Links undersampling algorithms are combined for data balancing. Thanks to this combination, the disadvantages of both oversampling and undersampling techniques are eliminated. As a result, the classification performance of the intrusion detection system has been improved and the performance improvement has been confirmed by the results obtained.

6) The proposed method is compared with nine different machine learning and deep learning classification techniques. The results showed that the proposed method outperforms the current and hybrid methods in the literature.

The remainder of the work is organized as follows. Related studies are mentioned in Chapter 2.

Chapter 3 provides general information about WSN-specific DoS attacks and Intrusion Detection Systems. In Chapter 4, the proposed DoS intrusion detection system is mentioned. The evaluation parameters and experimental results are shown in Chapter 5. Finally, in Chapter 6, the results obtained and future work are mentioned.

II. RELATED WORKS

With the emergence and widespread application of WSNs, traditional IDS solutions designed for wired networks have fallen short. Therefore, there is a need to design IDSs suitable for the structure and constraints of WSNs. Anomaly-based intrusion detection systems consider any deviation from normal behavior as an attack. According to the structure and characteristics of WSNs, some effective anomaly detection methods are suggested in the literature, including classification algorithms, clustering algorithms, machine learning algorithms and statistical learning models.

Almomani *et al.* [6] created a customized dataset called WSN-DS for WSN networks. An Artificial Neural Network has been trained on this dataset and different DoS attacks have been successfully classified. Vinayakumar *et al.* [7] developed an IDS for attack detection and classification. They proposed a scalable DNN framework called Scale-Hybrid-IDS-AlertNet against network attacks. The proposed method has been tested on NSL-KDD, UNSW-NB15, WSN-DS and CICIDS2017 datasets. Ioannou *et al.* [8] proposed an anomaly-based intrusion detection system called mIDS, which uses Binary Logistic Regression(BLR) statistical tools to classify sensor behaviors as good or bad. BLR model can only do binary classification. The proposed model has an accuracy rate of 91%. Le *et al.* [9] implemented a Random Forest algorithm to classify DoS attacks on the WSN-DS dataset. The performances of Random Forest and ANN algorithms were compared. It is stated that the Random Forest algorithm gives better results than ANN. Mahbooba *et al.* [10] proposed AI-based approaches for intrusion detection. In the study, the performances of machine learning and deep learning approaches in intrusion detection were compared. One- and two-layer LSTM networks were used as a deep learning approach. Two datasets, WSN-DS and KDD Cup network attack dataset, were used to classify the proposed approaches.

Jiang *et al.* [11] proposed an intrusion detection system designed for WSNs called SLGBM. In the study, feature selection was made using the sequence backward selection (SBS) algorithm to reduce the data size. LightGBM algorithm is used to classify different DoS attacks. The proposed method has been tested on the WSN-DS dataset. The proposed method has shown very successful results in detecting and classifying attacks. Liu *et al.* [12] proposed a network intrusion detection system based on adaptive synthetic (ADASYN) oversampling technology and LightGBM. Data imbalance was also discussed in the study. The proposed method was tested on the NSL-KDD, UNSW-NB15 and CICIDS2017 datasets and showed accuracy performance of 92.57%, 89.56% and 99.91%, respectively. Yao *et al.* [13] proposed a feature engineering based AutoEncoder(AE)-LightGBM intrusion detection system for SDN. The proposed system first uses Borderline-SMOTE to optimize data distribution, then AE is used for feature engineering to extract key features. Finally, LightGBM is trained to detect attacks using extracted features. The proposed method has been tested on KDDCup99 and NSL-KDD datasets. Ismail *et al.* [14] presented a comparative study and performance analysis of different machine learning classification techniques for the detection of cyber attacks in WSNs. They investigated the performance of three techniques: GBM, LightGBM, and Catboost. Performances were compared with three machine learning methods, Gaussian NB, KNN and RF. Feature selection and size reduction processes were also performed using the WSN-DS dataset in the study. Ismail *et al.* [15] presents a lightweight, multi-layered machine learning detection system to mitigate cyberattacks targeting WSNs. The multi-layer detection system consists of monitor nodes and two machine learning models deployed in the Base Station (BS). A Naive Bayes algorithm is used for binary classification in the first layer and a LightGBM algorithm is used for multiclass classification in the second layer. The proposed system was able to detect four DoS attacks observed in the WSN-DS dataset.

Ashwini and Manivannan [16] compared the performance of different machine learning algorithms on the NSLKDD dataset for intrusion detection. Al and Dener [17] presented a hybrid deep learning approach for intrusion detection. In addition, the problem of data imbalance is also addressed in the study. The proposed method has been tested on CIDDS-001 and UNSW-NB15 datasets. The proposed method has shown very successful results in detecting and classifying attacks. Souza *et al.* [18] proposed the hybrid DNN-kNN hybrid method on NSL-KDD and CICIDS2017 datasets for IoT security. The proposed approach reached 99.77% accuracy in the NSL-KDD dataset and 99.85% in the CICIDS2017 dataset. In another study on IoT attacks [19], a deep learning approach was suggested by Susilo and Sari against DoS attacks. Liu *et al.* [20] proposed another intrusion detection system for IoT. In the proposed work, a particle swarm optimization-based gradient descent (PSO-LightGBM) is proposed for intrusion detection. In the study, PSO-LightGBM was used to extract the features of the data and the extracted features were given as input to one-class SVM (OCSVM). The UNSW-NB15 dataset was used to validate the proposed intrusion detection model. Tang *et al.* [21] proposed an intrusion detection system based on LightGBM and AE. The proposed LightGBM-AE model consists of three steps: data preprocessing, feature selection and classification. The LightGBM-AE model uses the LightGBM algorithm for feature selection, then an autoencoder for training and detection. The proposed method has been tested on the NSL-KDD dataset. Alqahtani *et al.* [22] proposed a new intrusion detection system based on a genetic algorithm and extreme gradient boosting (XGBoot) classifier, called the GXGBoost model. In the study, the data imbalance problem is also discussed for performance improvement. The proposed method has been tested on the WSN-DS dataset. Tan *et al.* [23] proposes a method that uses a synthetic minority oversampling technique (SMOTE) to balance the dataset and then uses the random forest algorithm for attack detection. In the study, it was stated that the data balancing process increased the classification accuracy. Ifzarne *et al.* [24] designed a WSN-specific intrusion detection system. The proposed model is based on Information Gain Ratio and an online Passive aggressive classifier. First, Information Gain Ratio is used to select the relevant properties of the sensor data. Second, the online Passive aggressive algorithm is trained to detect and classify different types of DoS attacks. Studies were carried out on the WSN-DS dataset. A system has been proposed by Yadak and Kumar [25] to detect and prevent distributed denial-of-service attacks in wireless sensor networks. A Recurrent neural network is used as a classifier in the proposed model. The algorithm was tested on the WSN-DS dataset and achieved a success rate of 99.8%.

Pan *et al.* [43] proposed a lightweight intrusion detection model for WSNs. The proposed algorithm combines the k-nearest neighbor algorithm (kNN) and the sine cosine algorithm (SCA). The proposed algorithm significantly increased the classification accuracy and significantly reduced the false positive rate. The proposed algorithm has been tested on NSL-KDD and UNSW-NB15 datasets. With the proposed method, an accuracy performance of 99.33% for NSL-KDD and 98.27% for UNSW-NB15 was obtained. Zamry *et al.* [44] designed a lightweight anomaly detection system that reduces computational complexity and memory usage while providing high accuracy. One-class learning and dimension reduction concepts were used in the study. The One-Class Support Vector Machine (OCSVM) was used for one-class learning and the Candid Covariance-Free Incremental Principal Component Analysis (CCIPCA) algorithm was used for size reduction. The proposed system achieved 98.56% accuracy performance. Tabbaa *et.al.* [45] proposed a method for detecting Blackhole, Grayhole, Flooding, and Scheduling attacks. An ensemble method consisting of Adaptive Random Forest (ARF) and Hoeffding Adaptive Tree (HAT) algorithms was used in the study. The proposed method has been tested on the WSN-DS dataset. Mittal *et al.* [46] proposed an SVM-based anomaly detection system. NSL-KDD dataset was used in the study and 96.15% accuracy performance was obtained.

As can be seen from related studies, most of the studies have focused on classification-based attack detection. Many of the proposed approaches are datasets that contain both traditional network-specific data and are outdated. Besides, most of the studies that propose an intrusion detection system have not addressed the problem of data imbalance very much. In addition, feature selection has been ignored in most studies. In this study, a comparison of machine learning and deep learning approaches in WSN-specific intrusion detection systems has been made. In addition, the effects of data balancing and feature selection techniques on intrusion detection performance were evaluated. In addition to these, all the proposed work is carried out in a big data environment to highlight the need for big data environments due to the increasing volume of WSNs data day by day.

TABLE 1. Comparison of other works on intrusion detection.

Year	Authors	Model	Dataset	Features	Accuracy(%)
2016	Almomani et al [6]	ANN	WSN-DS	23	98.53
2017	Ioannou et al [8]	BLR	Own dataset	5	91.00
2018	Le et al.[9]	Random Forest	WSN-DS	23	98.00
2019	Vinayakumar et al [7]	DNN	KDD Cup'99 NSL-KDD UNSW-NB15 CICIDS2017 WSN-DS	41 41 49 79 23	95.00-99.00 95.00-99.00 65.00-75.00 93.00-96.00 96.00-99.00
2019	Alqahtani et al.[22]	GXGBoost	WSN-DS	23	99.70
2019	Tan et al.[23]	Random Forest	KDD Cup'99	41	92.57
2020	Jiang et al.[11]	LightGBM	WSN-DS	23	99.73
2020	Ashwini and Manivannan.[16]	LightGBM	NSL-KDD	41	78.00
2020	Souza et al.[18]	DNN, kNN	NSL-KDD CICIDS2017	41 79	99.77 99.85
2020	Susilo and Sari[19]	CNN	BoT-IoT	43	91.15
2020	Tang et al.[21]	LightGBM-AE	NSL-KDD	41	89.82
2020	Ifzarne et al.[24]	ID-GOPA	WSN-DS	23	95.69
2021	Mahbooba et al [10]	LSTM	WSN-DS	23	99.88
2021	Liu et al.[12]	LightGBM	NSL-KDD UNSW-NB15 CICIDS2017	41 49 79	92.57 89.56 99.91
2021	Yao et al.[13]	AE-LightGBM	KDDCup'99 NSL-KDD	41 41	99.90 99.70
2021	Ismail et al.[14]	LightGBM	WSN-DS	23	99.30
2021	Al and Dener.[17]	CNN-LSTM	CIDDS-001 UNSW-NB15	14 49	99.83 99.17
2021	Liu et al.[20]	PSO-LightGBM	UNSW-NB15	49	86.68
2021	Pan et al.[43]	kNN-SCA	NSL-KDD UNSW-NB15	41 49	99.33 98.27
2021	Zamry et al.[44]	OCSVM-CCIPCA	IBRL, LUCE, PDG, and NAMOS	-	98.56
2021	Mittal et al.[46]	SVM	NSL-KDD	41	96.15
2022	Ismail et al.[15]	NB-LightGBM	WSN-DS	23	99.30
2022	Yadav ve Kumar[25]	RNN	WSN-DS	23	99.80
2022	Tabbaa et al.[45]	ARF-HAT	WSN-DS	23	96.84
2022	Proposed Model	LigthGBM	WSN-DS	23	99.95

Table 1 presents relevant studies focusing on intrusion detection using deep learning and machine learning algorithms based on models, datasets, features, and accuracy parameters.

Since KDD Cup'99 and NSL-KDD datasets are out of date, UNSW-NB15, CIDDS-001 and CICIDS2017 datasets have been used frequently in recent years. Although these datasets are not created specifically for WSNs, they are also used in both intrusion detection systems designed for WSNs and intrusion detection systems designed for traditional networks. For these reasons, the WSN-DS dataset was used in this study due to more recent attacks, a greater amount of data and being specific to WSNs.

III. DoS ATTACKS AND DoS INTRUSION DETECTION SYSTEM FOR WSNs

In this section, information about DoS attacks specific to WSNs and DoS Intrusion Detection Systems that enable them to be detected successfully are presented. In the study, DoS intrusion detection systems are named DDS.

A. WSNs

A sensor network is an infrastructure of sensing, computation and communication elements that gives the ability to display, observe and react to events in a given environment [26]. The environment perceived by sensor networks can be the physical world, a biological system, or an information technology (IT) environment. Typical applications of sensor networks include data collection, monitoring, surveillance and medical telemetry. In addition to sense, it can often be done with wireless sensor networks in control and activation related applications. The sensors in WSNs have various purposes, functions and capabilities.

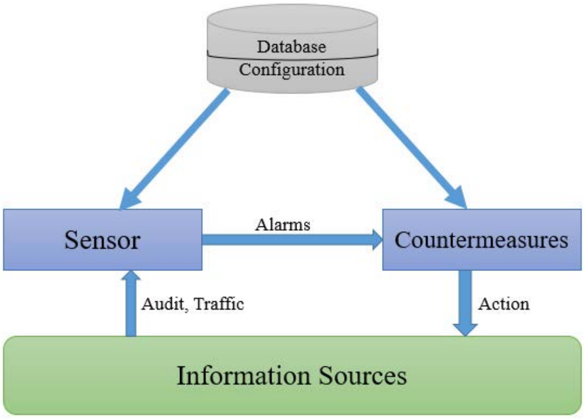


FIGURE 2. Basic IDS structure.

This area attracts great attention with the rapidly developing technology and the increase in potential application areas. WSNs are a multidisciplinary field that includes sensor networks, radio signals and network infrastructure, signal processing, artificial intelligence, database management, system architectures for operator-friendly infrastructure management, resource optimization, power management algorithms, and platform technology (such as hardware and software) [28], [29].

B. DoS ATTACKS IN WSNs

One of the most common attacks on WSNs that can target all layers of the protocol stack is the DoS attack. DoS attacks target the accessibility of information and information systems. The main purpose of these attacks is to disrupt the functioning of the network by blocking the services provided by the sensor nodes. Attackers prevent network nodes from using their resources with various types of attacks. Decrease in network performance, unresponsiveness of some parts of the network, increase in spam messages, delay or loss of packets can be indicators of DoS attack [30]. There are many different types of DoS attacks according to each layer and protocol specific to WSNs.

C. INTRUSION DETECTION SYSTEMS(IDS) FOR WSNs

Intrusion detection systems are generally divided into two groups according to the detection method: signature-based and anomaly-based. In a signature-based system, attackers are detected from previously known attacks. In anomaly-based systems, attacks are detected from the unusual behavior of the systems. An anomaly-based IDS approach is presented in this study. The basic IDS structure is shown in Fig. 2.

In WSNs, IDSs should be installed in places with more resources, such as base stations, where sensor nodes can be monitored in order to defend against threats to the network. The IDS structure specific to WSNs is shown in Fig. 3.

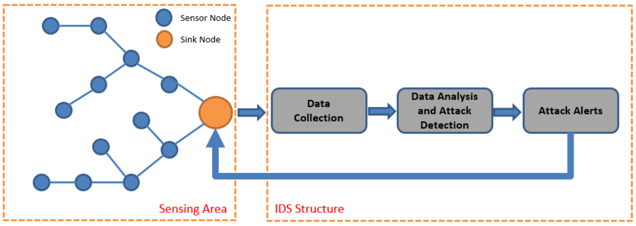


FIGURE 3. IDS scheme for WSN.

IDSs have three basic components: data collection, analysis-detection and alarming. The data collection component is used to monitor the node itself or neighboring nodes. The main component of IDSs is the analysis and detection component, which is responsible for detecting network behavior and activities on it and then analyzing them to decide if there is any abnormal behavior. The alarm component is responsible for alerting administrators when an intrusion is detected.

IV. PROPOSED METHOD

In this study, a new DoS intrusion detection system called STLGBM-DDS is proposed. The main purpose of the proposed system is to detect DoS attacks specific to WSNs, the use of which is increasing day by day, interacting with each other more and the network size is growing. For this purpose, the LightGBM machine learning algorithm is combined with feature selection and data imbalance processing in the proposed system. The proposed system consists of data preprocessing, data splitting, data balancing, classification and evaluation sections as shown in Fig. 4. In the data preprocessing stage, the raw dataset is made ready for classification algorithms. In addition, with the feature selection in the data preprocessing stage, the feature size is adjusted to maximize the algorithm performance. In the dataset splitting phase, the dataset is divided into two, a training dataset and a test dataset in accordance with training and testing purposes. In the data balancing phase, oversampling is done by resampling the data with SMOTE according to the minority class. Then, Tomek-Links undersampling approach was used in order to avoid the overfitting problem and to reduce noise in newly produced data. At this stage, with the combination of SMOTE and Tomek-Links methods called STL (SMOTE + Tomek-Link), the performance of the intrusion detection system is increased, while the dataset becomes balanced. In the classification phase, attacks are classified using the LightGBM machine learning method. Finally, the results obtained in the evaluation phase are evaluated according to the evaluation parameters and the performance of the method is determined. With the proposed method, DoS attacks are detected with high accuracy by balancing significantly imbalanced WSN data.

A. DATASET

WSN-DS is a dataset created specifically to detect attacks on WSNs. The WSN-DS dataset was created by Almomani *et al.* [6] to help better detect and classify DoS attack types.

While creating the dataset, the LEACH protocol was used because it is one of the most common and frequently used routing protocols in WSNs. The ns-2 simulation environment was used to collect the data. The dataset contains 23 features extracted using the LEACH routing protocol. The LEACH protocol is a routing protocol that uses 23 attributes to describe the state of each sensor node in the wireless network. These 23 features are: Id, Time, Is_CH, who_CH, RSSI, Dist_To_CH, M_D_CH, A_D_CH, Current Energy, Consumed Energy, ADV_S, ADV_R, JOIN_S, JOIN_R, ADV_SCH_S, ADV_SCH_R, Rank, DATA_S, DATA_R, Data_Sent_BS, Dist_CH_BS, Send_code, Attack_Type. However, only 19 features are included in the dataset file along with the class label as shown in Table 2.

The number of samples in the WSN-DS dataset is 374.661. The WSN-DS dataset includes 4 attack types. The samples in the dataset are labeled into five different classes: Blackhole, Grayhole, Flooding, TDMA and Normal, four of which are DoS attack types.

The samples in the dataset are labeled into five different classes, four of which are DoS attack types. Table 3 shows the detailed data distribution by class. In addition, Table 4 shows the count, mean, std, min and max values of the dataset. In Table 4, the ID attribute has been omitted only because it is used as an identifier for sensors and is meaningless in intrusion detection. Also, the Attack Type attribute has been omitted because it is categorical. For this reason, 17 features were carried out in the next stages of the study.

In this study, all evaluations were made on the WSN-DS dataset. The dataset was chosen because it contains DoS attacks specific to WSNs. In addition, the dataset has been preferred because it is up-to-date and has been used a lot in machine learning studies in recent years.

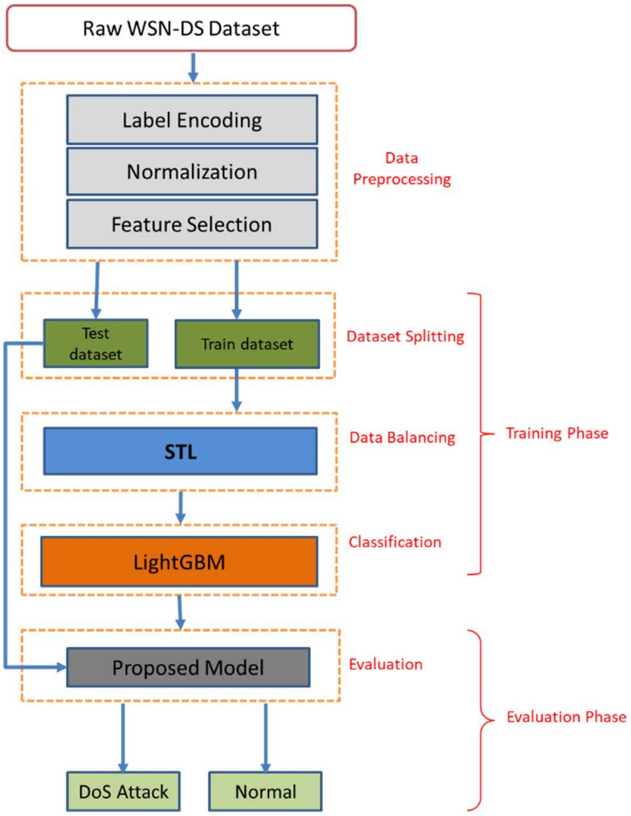


FIGURE 4. A schematic diagram of proposed mode.

B. NORMALIZATION AND ENCODING

In this study, before applying the classification algorithms to the dataset, the categorical values in the dataset were assigned to numerical values with the One-Hot Encoding process, and then the normalization process shown in Equation 1 was performed. As a result of the normalization process, all the numerical values in the data set were converted to a value between 0 and 1.

$$x' = (x - \mu)/\sigma \tag{1}$$

where x is the original value, x′ is the normalized value, μ and σ are the mean and standard deviation values, respectively. Thanks to the normalization process, some features with high numerical values are prevented from affecting the algorithm result and negatively affecting the performance. In the WSN-DS dataset, the Attack_Type property consists of textual expressions such as Normal, Grayhole, Blackhole, TDMA and Flooding. These textual expressions prevent the calculations of artificial intelligence algorithms. Therefore, these expressions need to be converted to numeric values. The classification labels were converted to the values seen in Table 5 as a result of One-Hot Encoding.

C. FEATURE SELECTION

Feature selection is a technique that removes irrelevant and redundant features and selects the most optimal subset of features. Feature selection is necessary and important for

TABLE 2. Detailed description of the attributes of the WSN-DS dataset.

No	Attribute Name	Attribute Description
1	Id	It is a unique ID to distinguish the sensor node in any round and at any stage.
2	Time	It is the current simulation time of the sensor node.
3	Is_CH	It is a flag to distinguish whether the node is CH, or not. Value 1 means CH and value 0 means normal node.
4	who_CH	It is a ID of the CH in the current round
5	Dist_To_CH	It is the distance between the node and its CH in the current round
6	ADV_S	It is the number of advertise CH's broadcast messages sent to the nodes
7	ADV_R	It is the number of advertise CH messages received from CHs
8	JOIN_S	It is the number of join request messages sent by the nodes to the CH.
9	JOIN_R	It is the number of join request messages received by the CH from the nodes
10	SCH_S	It is the number of advertise TDMA schedule broadcast messages sent to the nodes
11	SCH_R	It is the number of TDMA Schedule messages received from CHs
12	Rank	It is the order of this node within the TDMA schedule.
13	DATA_S	It is the number of data packets sent from a sensor to its CH.
14	DATA_R	It is the number of data packets received from CH.
15	Data_Sent_BS	It is the number of data packets sent to the BS
16	Dist_CH_BS	It is distance between the CH and the BS.
17	Send_code	It is the cluster sending code
18	Consumed_Energy	It is the energy amount consumed by the sensor node in the previous round
19	Attack_Type	Type of Attack (Blackhole, Grayhole, Flooding, TDMA, Normal)

TABLE 3. The number of samples in each class of WSN-DS dataset.

Class	Number of Samples	Proportion(%)
Normal	340066	90.77
Grayhole	10049	2.68
Blackhole	14596	3.90
TDMA	3312	0.88
Flooding	6638	1.77
Total	374661	100

TABLE 4. Statistical analysis of WSN-DS dataset.

Attribute Name	Count	Mean	Std	Min value	Max value
Time	374661	1064.749	899.6462	50	3600
Is_CH	374661	0.115766	0.319945	0	1
who_CH	374661	274980.4	389911.2	101000	3402100
Dist_To_CH,	374661	22.59938	21.95579	0	214.2746
ADV_S	374661	0.267698	2.061148	0	97
ADV_R	374661	6.940562	7.044319	0	117
JOIN_S	374661	0.779905	0.414311	0	1
JOIN_R	374661	0.737493	4.691498	0	124
SCH_S	374661	0.288984	2.754746	0	99
SCH_R	374661	0.747452	0.434475	0	1
Rank	374661	9.687104	14.6819	0	99
DATA_S	374661	44.85792	42.57446	0	241
DATA_R	374661	73.89004	230.2463	0	1496
Data_Sent_BS	374661	4.569448	19.67916	0	241
Dist_CH_BS	374661	22.56274	50.2616	0	201.9349
Send_code	374661	2.497957	2.407337	0	15
Consumed_Energy	374661	0.305661	0.669462	0	45.09394

TABLE 5. Name-number matching of classes.

0	Normal
1	Grayhole
2	Blackhole
3	TDMA
4	Flooding

machine learning and deep learning processes, as sometimes irrelevant features affect the performance of models.

Besides, feature selection can save storage space, increase computation speed by reducing computational load, remove unnecessary features, reduce noise and avoid the overfitting problem. Feature selection processing becomes even more important for WSNs with limited resources, as it alleviates the energy requirement and computational burden. Therefore, feature selection is an important component in WSN-specific IDS design.

In this study, Pearson Correlation Coefficient and Information Gain Ratio feature selection methods were used to observe the correlation between features and feature selection. The Pearson correlation matrix shown in Fig. 5 was used as feature analysis to observe the relationships of each feature in the WSN-DS dataset with other features in the dataset. Pearson Correlation Coefficient refers to test statistics that measure the statistical relationship between two continuous variables. As another definition, it is a measure of linear correlation between two data sets [31]. Since it is based on the covariance method, it is known as the best method of measuring the relationship between the variables of interest. It gives information about the size of the relationship or the direction of the relationship as well as the correlation. It always produces results with a value between −1 and 1. It essentially refers to a normalized measure of covariance. It is formulated as:

$$r = \frac{n\left(\sum xy\right) - \left(\sum x\right)\left(\sum y\right)}{\sqrt{\left[n\sum x^2 - \left(\sum x\right)^2\right]\left[n\sum y^2 - (y)^2\right]}} \tag{2}$$

Here,

- r = Pearson Coefficient
- n = number of the samples

- $\sum$xy = sum of products of the paired samples
- $\sum$x = sum of the x scores
- $\sum$y = sum of the y scores
- $\sum$x2 = sum of the squared x scores
- $\sum$y2 = sum of the squared y scores

Information Gain Ratio is defined as the ratio of information gain to the intrinsic knowledge [32]. The Information Gain Ratio was proposed by Ross Quinlan [33] to reduce the Information Gain's bias towards features with a large diversity value. It is formulated as:

$$\boldsymbol{GR}\,(\boldsymbol{T}) = \boldsymbol{IG}(\boldsymbol{T})/\boldsymbol{H}(\boldsymbol{T}) \tag{3}$$

where GR(T) is the information gain ratio of the T feature, IG(T) is the information gain of the T feature and H(T) is intrinsic information value of T. The gain ratio takes into account the number and size of branches when selecting an attribute and corrects the information gained by taking into account the intrinsic knowledge of a split. Inside information is the ignoring of information about the class [24]. The data obtained as a result of the Information Gain Ratio are shown in Table 6.

As can be seen in Fig. 5, although many features in the WSN-DS dataset do not have high correlations, it is seen that the Id and who_CH features have high correlations among themselves. Therefore, as a result of this operation, the Id attribute was removed from the dataset. Then, Information Gain Ratio was used as a second method to increase efficiency and accuracy in the selection of features.

As can be seen from Table 6, the features with the lowest impact on the class are Time, who_CH, Id, DATA_R ve dist_CH_To_BS, respectively. Therefore, these features were excluded from the dataset.

As a result, feature selection, which is one of the data preprocessing steps, was made in order to increase the performance of the proposed IDS in this study. As a result of the feature selection process, the number of features has been reduced so that the WSN-DS dataset has 13 features.

D. DATA BALANCING

The WSN-DS dataset is a dataset of imbalanced classes as shown in Fig. 6. The unbalanced distribution among the classes negatively affects the classification performance. Especially, minority classes affect the detection rate negatively [17]. The Imbalanced class problem is not adequately considered in intrusion detection system design. Using only undersampling techniques for data imbalance results in the elimination of useful normal network traffic significantly reduces the amount of data used for training purposes. Using oversampling techniques alone causes unnecessary data size increase and noise. In this study, an approach called STL (Smote + Tomek-Link) Synthetic Minority Oversampling Technique (SMOTE) and Tomek-Links oversampling and undersampling approaches are proposed to overcome the imbalanced class problem [17].

TABLE 6. Information gain ration results for WSN-DS dataset.

Attribute Name	Feature No	p-value
ADV_S	6	0.6736
Is_CH	3	0.6733
SCH_S	10	0.4644
DATA_S	13	0.4331
JOIN_R	9	0.2970
JOIN_S	8	0.2888
Dist_To_CH	5	0.2886
send_code	17	0.2257
SCH_R	11	0.2205
Rank	12	0.1634
Data_Sent_To_BS	15	0.1158
ADV_R	7	0.0620
ConsumedEnergy	18	0.0575
dist_CH_To_BS	16	0.0454
DATA_R	14	0.0399
Id	1	0.0291
whoCH	4	0.0286
Time	2	0.0213

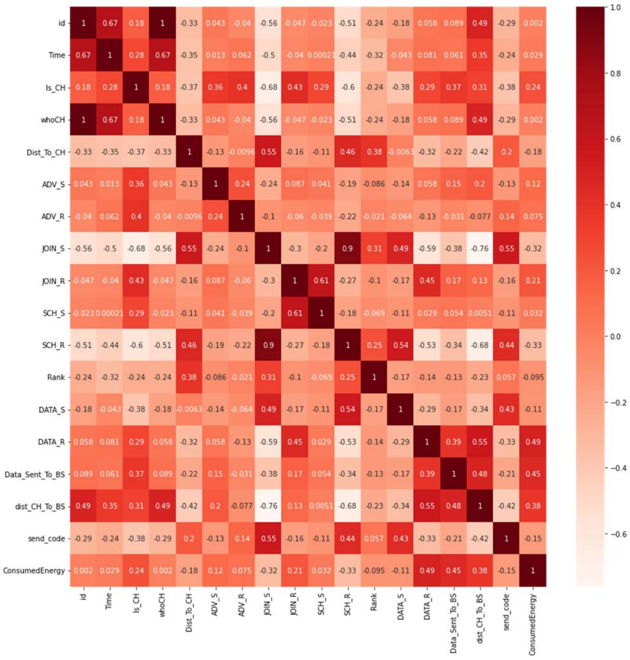
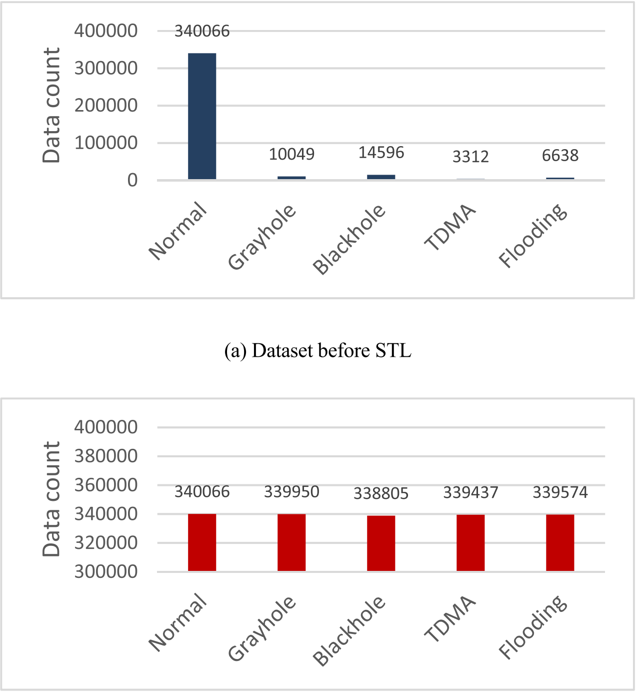


FIGURE 5. Pearson correlation matrix for WSN-DS datase.

1) SYNTHETIC MINORITY OVERSAMPLING TECHNOLOGY(SMOTE)

SMOTE is a heuristic oversampling technique proposed by Chawla *et al.* [34] to solve the problem of class imbalance in datasets. In this method, synthetic data is produced by oversampling the data in the minority class. SMOTE also overcomes the overfitting problem caused by random oversampling methods by generating synthetic data. It has been widely used in the field of class imbalance in recent years, as it significantly improves the overfitting situation caused by the non-heuristic random sampling method [23]. SMOTE increases the number of minority class samples by adding randomly generated new samples between minority class samples and their neighbors and improves the class



(a) Dataset before STL

(b) Dataset after STL

FIGURE 6. **WSN-DS dataset a) before data balancing b) after data balancing.**

imbalance problem [35], [36], [37]. SMOTE works by selecting samples close to the feature space. A random sample is chosen from the Minority class, and then k nearest neighbors of the selected sample are found. After randomly choosing one of the nearest neighbors, the difference between the two sample features is multiplied by a number between 0 and 1 and added to the selected sample value. A line is then drawn between the two sample features and synthetic samples are produced along this line. Based on the amount of oversampling required, neighbors from the k nearest neighbors are randomly selected.

SMOTE samples are linear combinations of two similar samples (S, s^R) from the minority class and are defined as follows:

$$n = s + d \cdot \left(s^R - s\right), \quad 0 \leq d \leq 1 \tag{4}$$

where s^R is the randomly selected sample of s according to the nearest neighbor number and d is the difference between the two samples.

2) TOMEK-LINKS

Tomek-Links is an undersampling technique applied to imbalanced datasets developed by Tomek. It can be considered as an improved version of the Nearest Neighbor Rule. In this approach, the samples on the Tomek link are removed from the dataset. It creates data sample pairs that are closest to each other in the dataset but belong to different classes. These data pairs are called Tomek links. The basic idea is to separate the minority and majority classes from each other.

Let x be an instance of one class and y an instance of another class, x and y are the nearest neighbors and d(x,y); provided that the distance between x and y is;

$$\begin{aligned} &T\,(x, y)\ \ is\ a\ Tomek - Link,\ \ if\ for\ any\ instance\ i, \\ &\quad \times d\,(x, y) < d\,(x, i)\ ord(x, y) < d(y, i) \end{aligned} \tag{5}$$

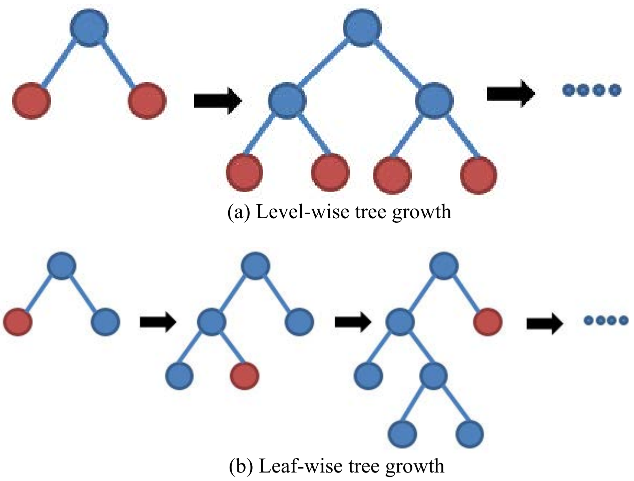
T-links separate the two classes. Data samples on this link are considered noise. Deleting majority class noises increases the class separation and stabilizes the data distribution. It should be noted here that the noise samples are deleted from the majority class. Fig. 6 shows the dataset resulting from the Tomek-Link undersampling process.

E. LightGBM CLASSIFICATION MODEL

Light Gradient Boosting Machine (LightGBM) is a free and open source distributed gradient boosting framework for machine learning applications developed by Microsoft. LightGBM is a gradient boosting framework that uses a fast, distributed and high-performance tree-based learning algorithm [38]. The size of the data produced through various information systems is increasing day by day. While this situation reveals the necessity of fast processing of data, it becomes difficult for traditional data science algorithms to give faster results. LightGBM is named Light because of its high speed. Thanks to this feature, it can process large data quickly and requires less memory. Another important feature of LightGBM is its focus on the accuracy of the results produced. LightGBM supports GPU learning and therefore data scientists widely use LGBM for data science application development [39].

Another advantage of LightGBM is that it supports the optimal division of categorical features. LightGBM supports the optimal separation of categorical features by a grouping method [40]. In this way, sparse data caused by numerical transformation is avoided. In addition to these advantages, it is an important disadvantage that it is sensitive to the overfitting problem in small-sized datasets.

Fig. 7 shows the difference between LightGBM from other tree-based algorithms. While other algorithms grow trees horizontally (level-wise), LightGBM grows the tree vertically (leaf-wise). The leaf with maximum delta loss is selected for the growth of the tree structure. When growing the same leaf, a leaf-wise algorithm can reduce loss more than a level-wise algorithm. As can be seen in Fig. 7, LightGBM typically consists of fewer decision trees and fewer leaves per decision tree. This makes LightGBM time efficient. LightGBM consists of two algorithms, Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB). LightGBM adopts an advanced histogram algorithm for the feature selection of the decision tree. Here, while the number of features is reduced by the EFB algorithm, the number of samples in the training phase is reduced by the GOSS algorithm. These two algorithms form the features of LightGBM and are combined



(a) Level-wise tree growth

(b) Leaf-wise tree growth

FIGURE 7. Comparison of tree growth structure for LightGBM and other boosting algorithms.

to give it an edge over other Gradient Boosting Decision Tree (GBDT) frameworks such as XGBoost [41], pGBRT [42].

GOSS is basically a data downsampling technique. During the model training phase, samples with large gradients have a greater effect on information gain. Therefore, the GOSS algorithm downsamples the data samples, keeping samples with large gradients and randomly dropping those with small gradients, which is not helpful in information acquisition. Suppose we have an independent and identically distributed dataset of size n, $\{\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots\ldots, \boldsymbol{x_n}\}$. Here, each $\boldsymbol{x_i}$ represents s-dimensional vectors in the $\boldsymbol{x^s}$ space. In each gradient boosting iteration, the negative gradient of the loss function according to the output of the model is expressed as $\{\boldsymbol{g_1}, \boldsymbol{g_2}, \ldots\ldots, \boldsymbol{g_n}\}$. In the GOSS method, the training samples are sorted in descending order according to the absolute values of their gradients. Next, samples with larger gradients are retained and a subset of samples, A, is obtained. Cluster B is formed by random sampling from samples with smaller gradients. Thus, with the GOSS algorithm, the number of low-impact samples is reduced in each iteration, thereby increasing the estimation ability. The gradient of each sample shows the degree of error in the sample estimation for which trained in the previous round. O refers to the training data at a fixed node in the decision tree for the gradient calculation of each sample. The information gain of the j segmentation feature at the d segmentation point is shown in equation (6):

$$V_j(d) = \frac{1}{n_o}\frac{\left(\sum_{x_i \in A_l} g_i\right)^2}{n_l^j(d)} + \frac{1}{n_o}\frac{\left(\sum_{x_i \in A_r} g_i\right)^2}{n_r^j(d)} \tag{6}$$

Here,

$$\begin{aligned} x_i &\in A_l = x_i \leq d \\ x_i &\in A_r = x_i > d \\ n_o &= \sum I\left[x_i \in O\right], \\ n_l^j(d) &= \sum I\left[x_i \in O : x_{ij} \leq d\right] \\ n_r^j(d) &= \sum I\left[x_i \in O : x_{ij} > d\right] \\ A_l &= \{x_i \in A : x_{ij} \leq d\} \\ \mathrm{A_r} &= \{\mathrm{x_i} \in \mathrm{A} : \mathrm{x_{ij}} > \mathrm{d}\} \end{aligned}$$

For the GOSS algorithm, a denotes the proportion of larger gradient samples and b ∈ (0, 1-a) denotes the proportion of smaller gradient samples to be randomly selected. The values of a and b are predetermined. GOSS randomly samples these data samples with small gradients in the data distribution with a constant factor of ((1-a))/b. In this way, GOSS reduces the data size by keeping the accuracy high without changing the distribution of the original dataset too much. Thus, the final information gain is calculated by equation (7):

$$V_j(d) = \frac{1}{n_o}\frac{\left(\sum_{x_i \in A_l} g_i + \frac{1-a}{b}\sum_{x_i \in B_l} g_i\right)^2}{n_l^j(d)} + \frac{1}{n_o}\frac{\left(\sum_{x_i \in A_r} g_i + \frac{1-a}{b}\sum_{x_i \in B_r} g_i\right)^2}{n_r^j(d)} \tag{7}$$

Here,

$$\begin{aligned} B_l &= \{x_i \in B : x_{ij} \leq d\} \\ \boldsymbol{B_r} &= \{\boldsymbol{x_i} \in \boldsymbol{B} : \boldsymbol{x_{ij}} > \boldsymbol{d}\} \end{aligned}$$

As a result, to determine the split point, the information gain $(\boldsymbol{V_j(d)})$ of a smaller subset of data is calculated instead of the information gain of the entire dataset. As a result, the computational load is significantly reduced. EFB is mainly used for sampling data and effectively reducing the number of features. EFB aims to reduce the number of features without harming the accuracy rate and accordingly increase the efficiency of model training. EFB has two basic processing steps. These are creating bundles and combining features into the same bundle. High-dimensional data are often very sparse. In a sparse feaute domain, many features are mutually exclusive. EFB can safely collect exclusive features in a single feature. Thus, EFB combines sparse features to create denser features. If the two feautes are not completely mutually exclusive, the conflict ratio is used to measure the degree of non-mutual exclusion between the feautes. The two features can be combined without affecting the final accuracy when the value is small. EFB generates histograms with the same features as individual features from the feature bundles obtained. Accordingly, the complexity is reduced, the accuracy level is maintained, and the training process is faster with lower memory consumption.

V. EXPERIMENTS AND EVALUATIONS

In this section, the proposed method is implemented on the imbalanced WSN-DS dataset specific to WSNs. The studies were carried out using the Pyspark tool, which provides python programming language support on the Apache Spark big data platform in the Google Colab environment. For machine learning and deep learning algorithms, Scikit-learn and Keras libraries, which are included in the PySpark MLib library, were used, respectively. The proposed method was compared with nine different machine learning and deep learning algorithms, evaluations were made and the results were interpreted.

A. EVALUATION PARAMETERS

The most commonly used parameters in the literature such as Accuracy, Precision, F-score, Recall, ROC and Precision-Recall curves were used in the evaluation of the results. These evaluation parameters are used in many classification problems [47], [48]. These values are based on the comparison of the classification results obtained as a result of various machine learning or deep learning algorithms with the required classification values. In other words, these values are obtained by interpreting the results produced by the models. These parameters are derived from the confusion matrix data. The basic elements of the confusion matrix are true-positive (TP), true-negative (TN), false-positive (FP) and false-negative (FN). TP represents the number of instances correctly classified as an attack. TN represents the number of samples correctly classified as normal. FP refers to the misclassification of normal samples as attack samples. Similarly, FN refers to the misclassification of attack samples and accepting them as normal samples.

The accuracy parameter is defined as the ratio of all correctly classified samples (TP, TN) to all samples (TP, TN, FP and FN) and is represented by Equation 8. Precision is a metric that measures the number of correct positive (TP) predictions made and is expressed by Equation 10 [49]. Precision is the ratio of all correctly classified attacks (TP) to the number of correctly classified attacks (TP) and incorrectly classified normal samples (FP). Recall is a metric that measures the number of correct positive predictions made from all positive predictions that can be made. Of all positive predictions, Precision only comments on correct positive predictions, while Recall provides an indication of missed positive predictions. Recall is calculated by the ratio of the number of correctly classified positive samples to the number of all correctly classified samples and is shown by Equation 11 [49]. The harmonic mean of the Sensitivity and Recall parameters is known as the F-score and is calculated by Equation 9 [50]. F-Measure, which weights Precision and Recall equally, is one of the most frequently used variables when learning from data [51].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{8}$$

$$F - Score = \frac{2TP}{2TP + FP + FN} \tag{9}$$

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Recall = \frac{TP}{FN + TP} \tag{11}$$

TABLE 7. Hyperparameters of LightGBM.

Hyperparameters	Values
Boosting type	gbdt
Learning rate	0,1
Number of leaves(Num_leaves)	32
Number of estimators(N_estimators)	210
Random state	2
Min_chield samples	20
objective	multiclass

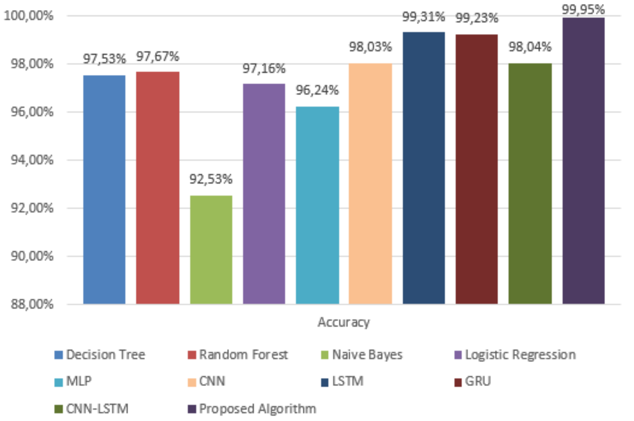


FIGURE 8. Comparison of accuracies.

TABLE 8. results of accuracy, precision and f-score parameters.

Models	Precision	Recall	F-Score	Accuracy
Decision Tree	0.97	0.98	0.97	97.53%
Random Forest	0.98	0.98	0.97	97.67%
Naive Bayes	0.88	0.93	0.90	92.53%
Logistic Regression	0.97	0.97	0.97	97.16%
MLP	0.96	0.96	0.96	96.24%
CNN	0.98	0.98	0.98	98.03%
LSTM	0.99	0.99	0.98	99.31%
GRU	0.99	0.99	0.98	99.23%
CNN-LSTM	0.98	0.98	0.98	98.04%
Proposed Algorithm	0.99	0.99	0.99	99.95%

B. RESULTS AND COMPARISON

The results obtained in the study were evaluated in three different aspects besides the general performance of the proposed method. First, the performance of the LightGBM classification algorithm used in the proposed method is compared against different traditional machine learning and deep learning algorithms. In addition, the performance of the LightGBM algorithm against the hybrid deep learning algorithms proposed in the literature was also evaluated. Secondly, the contribution of the STL data balancing algorithm used in the proposed method to the classification results was evaluated. Finally, the contribution of the feature selection process to the classification results was evaluated. In this study, the dataset is split into two, 70% for training and 30% for testing. First of all, generally accepted values are given for hyper parameters in deep learning algorithms and tuning

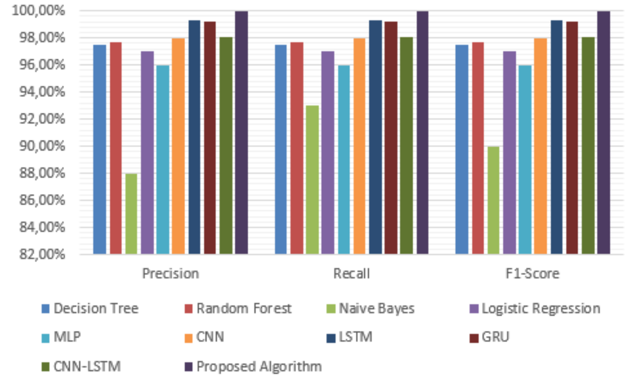


FIGURE 9. Comparison results according to Precision, Recall and F1-Score parameters.

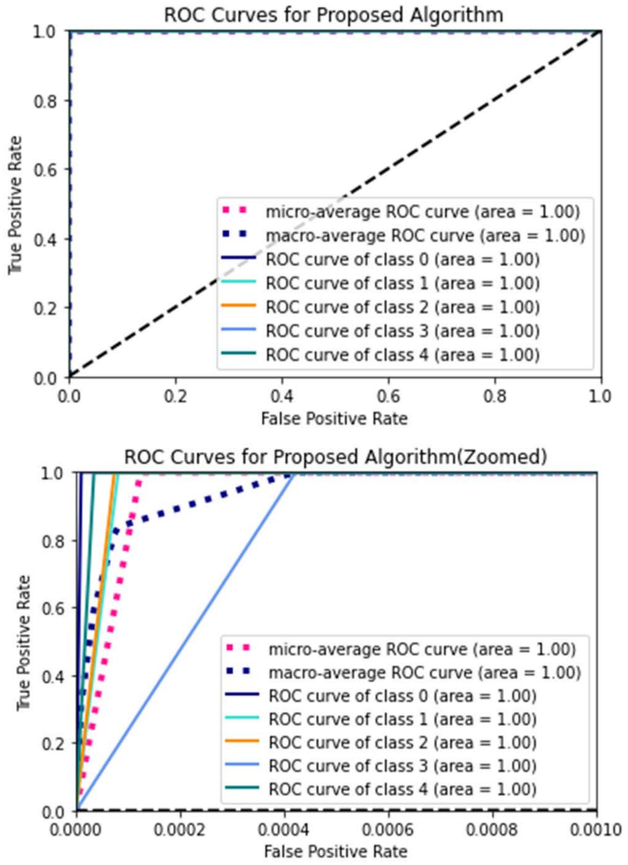


FIGURE 10. ROC curves for proposed algorithm.

is done for the best results. According to the result of the tuning process, the best results were obtained by using the hyper parameters shown in Table 7. Other hyper parameters of LightGBM are left at default values.

Fig. 8 presents the comparisons of the proposed algorithm and various machine learning and deep learning algorithms frequently used in the literature on the WSN-DS dataset, according to the accuracy parameter. As seen in Fig. 8, the proposed method gives the best accuracy result with 99.95% compared to the methods suggested in the literature. From the results obtained, it has been observed that deep learning algorithms achieve better results than traditional machine learning algorithms.

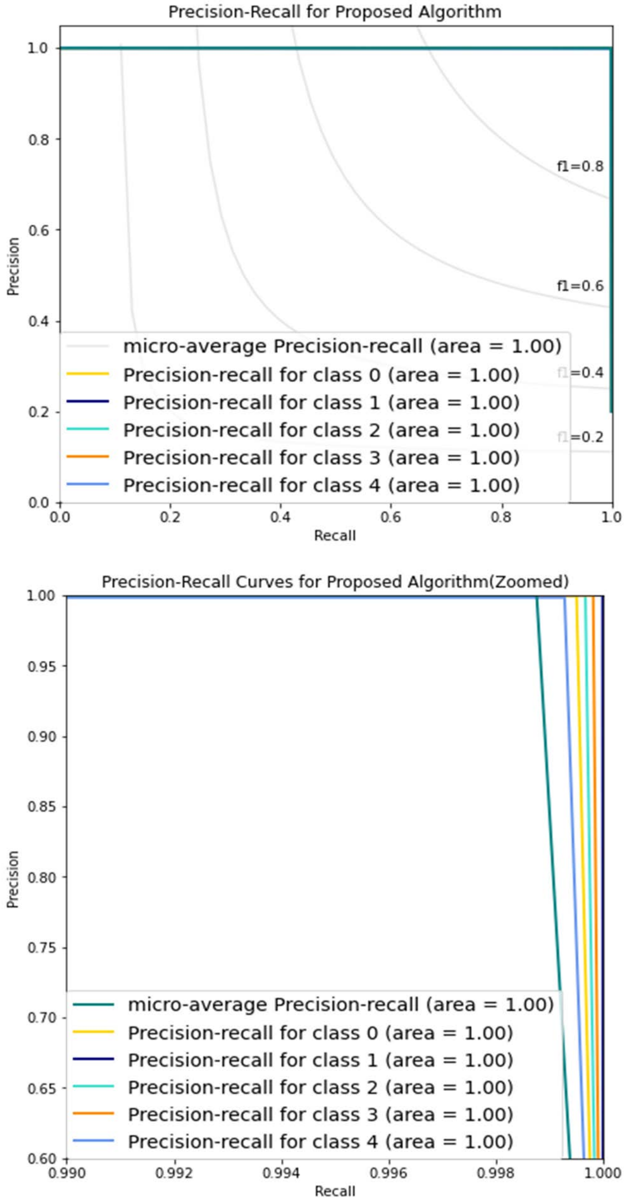
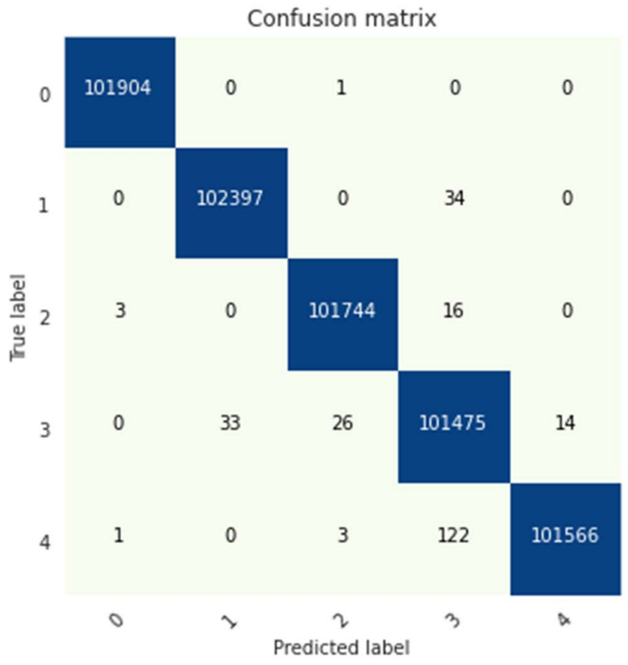


FIGURE 11. Precision-Recall curves for proposed algorithm.

TABLE 9. Classification performance of the proposed method.

	Normal	Grayhole	Blackhole	TDMA	Flooding
Accuracy	0.999990	0.999668	0.999813	0.999281	0.998761
Precision	0.999961	0.999678	0.999705	0.998308	0.999862
Recall	0.999990	0.999668	0.999813	0.999281	0.998761
F1-Score	0.999975	0.999673	0.999759	0.998794	0.999311

The remarkable point in the results obtained is that the CNN-LSTM hybrid approach individually performs worse



	0	1	2	3	4
0	101904	0	1	0	0
1	0	102397	0	34	0
2	3	0	101744	16	0
3	0	33	26	101475	14
4	1	0	3	122	101566

FIGURE 12. Confusion matrix.

than the CNN and LSTM deep learning algorithms on the WSN-DS dataset. In studies in the literature, hybrid methods generally produced more successful results than individual deep learning techniques, while individual methods performed better on the WSN-DS dataset. It is considered that this situation is caused by the unique feature structure of the WSN-DS dataset. The Naive Bayes algorithm showed the lowest performance. The Multinominal Naive Bayes algorithm was used in this study. The Naive Bayes algorithm showed the lowest performance, especially since it could not detect the data belonging to TDMA and Flooding classes at a high rate. A comparison of classification algorithms according to accuracy, precision, recall and F1-Score parameters is presented in Fig. 9 and Table 8. When the results of these parameters are examined, the proposed method shows the best results for each parameter. In addition, detailed classification results according to the evaluation parameters of the proposed method are presented in Table 9.

From the ROC and Precision-Recall curves shown in Fig. 10 and Fig. 11, respectively, it is seen that the proposed algorithm is quite successful for all classes.

The classification results of the proposed algorithm for each class are shown in Fig. 12 on the confusion matrix.

It is seen from the confusion matrix that the proposed algorithm is successful for all classes. Name matches of class numbers shown numerically in the figures are given in Table 5. In the study, the contribution of the STL algorithm used in the data imbalance processing stage, as another evaluation method, to the classification success was evaluated. Fig. 13 shows the effect of data balancing on DoS attack detection performance.

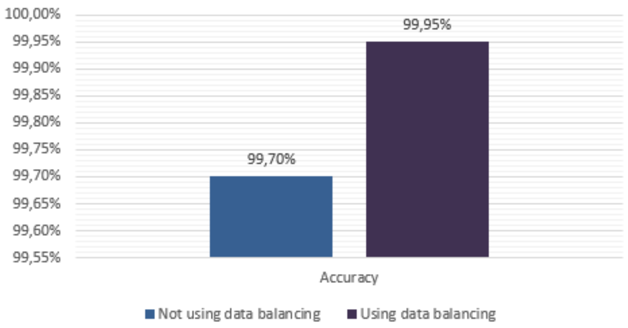


FIGURE 13. Data balancing effect on algorithm performance.

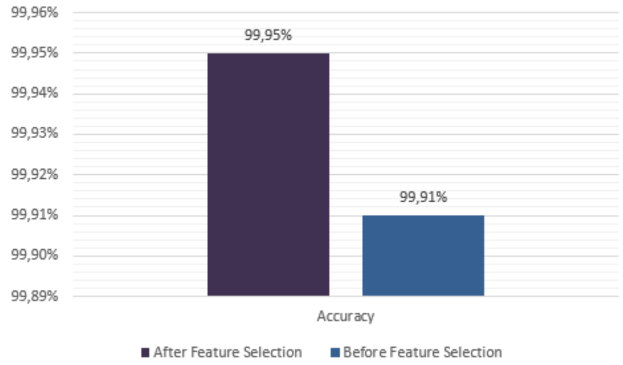


FIGURE 14. Effect of feature selection on DDS accuracy.

As can be seen from the figure, the correct detection rate of DoS attacks increased from 99.70% to 99.95%. Data balancing significantly improves the performance of the DoS intrusion detection system.

Finally, the effect of feature selection on algorithm performance is evaluated. As can be seen from Fig. 14, the feature selection process has an impact on the accuracy of DDS.

At this evaluation stage, the proposed algorithm without feature selection achieved 99.91% accuracy. As a result of the feature selection process, the performance of the proposed algorithm has increased to 99.95% accuracy. Although it is thought that it does not increase the accuracy rate numerically, in intrusion detection systems where each detection is important, the slightest increase in the correct detection rate is important. Because each attack can have important consequences.

VI. CONCLUSION AND FUTURE WORKS

In this study, a new classification-based DoS intrusion detection system is proposed to detect DoS attacks specific to WSNs. The proposed STLGBM-DDS approach combines LightGBM machine learning algorithm with data balancing and feature selection operations. In the study, the STL(SMOTE + Tomek-Link) ensemble algorithm was used for data balancing. LightGBM machine learning algorithm was used for the classification process. Experimental studies were performed on the WSN-DS dataset. All experimental studies were carried out on Apache Spark big data platform in

Google Colab environment. The performance of the proposed method was verified using the parameters Accuracy, Precision, Recall, F-Score, ROC curve and Precision-Recall curve. According to the experimental results, the proposed method performed better than the other methods in the literature with an accuracy value of 99.95%. In this study, the STL (SMOTE + Tomek-Links) algorithm, which consists of SMOTE oversampling and Tomek-Links undersampling methods, is used. In experimental studies, the effects of data balancing on classification performance were examined and the proposed method was verified. Within the scope of the study, the effects of feature selection on the WSN-DS dataset on the system performance were also evaluated. The WSN-DS dataset contains features with low correlation. This was observed using the Pearson Correlation Coefficient. In addition, using the Information Gain Ratio feature selection technique, features were ranked according to their effects and the number of features was reduced from 18 to 13. 5 features that did not affect the algorithm performance were excluded from the dataset. As a result of the feature selection process, it was observed that the algorithm performance increased. It was observed that the detection performance of the system decreased when more than 5 features were removed from the data set.

In the future, it is planned to combine the LightGBM machine learning algorithm with different machine learning and deep learning approaches such as CNN, LSTM, GRU and AE as a hybrid for performance improvement and evaluate the results. In addition, different oversampling and undersampling methods will be evaluated for data balancing. Besides, it is planned to evaluate the performance of the proposed method on different datasets. Finally, studies are planned to increase the reliability and transparency of intrusion detection systems with Explainable Artificial Intelligence (XAI) techniques.

REFERENCES

[1] H. Chen, P. Huang, M. Martins, H. C. So, and K. Sezaki, "Novel centroid localization algorithm for three-dimensional wireless sensor networks," in *Proc. 4th Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Oct. 2008, pp. 1–4.

[2] *Industrial Wireless Sensor Network Market Size, Share & Trends Analysis Report by Component (Hardware, Software, Service), by Type, by Technology, by Application, by End Use, and Segment Forecasts, 2020–2025*. Accessed: May 2, 2022. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/industrial-wireless-sensor-networks-iwsn-market

[3] O. A. Osanaiye, A. S. Alfa, and G. P. Hancke, "Denial of service defence for resource availability in wireless sensor networks," *IEEE Access*, vol. 6, pp. 6975–7004, 2018.

[4] M. Rassam, A. Maarof, and A. Zainal, "A survey of intrusion detection schemes in wireless sensor networks," *Amer. J. Appl. Sci.*, vol. 9, no. 10, pp. 1636–1652, 2012.

[5] I. Butun, S. D. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 266–282, 1st Quart., 2014, doi: 10.1109/SURV.2013.050113.00191.

[6] 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, Jan. 2016, doi: 10.1155/2016/4731953.

[7] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[8] I. Onat and A. Miri, "An intrusion detection system for wireless sensor networks," in *Proc. Int. Conf. Wireless Mobile Comput., Netw. Commun.*, 2017, pp. 1–5.

[9] T.-T.-H. Le, T. Park, D. Cho, and H. Kim, "An effective classification for DoS attacks in wireless sensor networks," in *Proc. 10th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2018, pp. 689–692, doi: 10.1109/ICUFN.2018.8436999.

[10] B. Mahbooba, R. Sahal, W. Alosaimi, and M. Serrano, "Trust in intrusion detection systems: An investigation of performance analysis for machine learning and deep learning models," *Complexity*, vol. 2021, pp. 1–23, Mar. 2021.

[11] S. Jiang, J. Zhao, and X. Xu, "SLGBM: An intrusion detection mechanism for wireless sensor networks in smart environments," *IEEE Access*, vol. 8, pp. 169548–169558, 2020, doi: 10.1109/ACCESS.2020.3024219.

[12] J. Liu, Y. Gao, and F. Hu, "A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM," *Comput. Secur.*, vol. 106, Jul. 2021, Art. no. 102289, doi: 10.1016/j.cose.2021.102289.

[13] R. Yao, N. Wang, Z. Liu, P. Chen, D. Ma, and X. Sheng, "Intrusion detection system in the smart distribution network: A feature engineering based AE-LightGBM approach," *Energy Rep.*, vol. 7, pp. 353–361, Nov. 2021.

[14] S. Ismail, T. T. Khoei, R. Marsh, and N. Kaabouch, "A comparative study of machine learning models for cyber-attacks detection in wireless sensor networks," in *Proc. IEEE 12th Annu. Ubiquitous Comput., Electron. Mobile Commun. Conf. (UEMCON)*, Dec. 2021, pp. 0313–0318, doi: 10.1109/UEMCON53757.2021.9666581.

[15] S. Ismail, D. Dawoud, and H. Reza, "A lightweight multilayer machine learning detection system for cyber-attacks in WSN," in *Proc. IEEE 12th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2022, pp. 481–486, doi: 10.1109/CCWC54503.2022.9720891.

[16] A. B. Abhale and S. S. Manivannan, "Supervised machine learning classification algorithmic approach for finding anomaly type of intrusion detection in wireless sensor network," *Opt. Memory Neural Netw.*, vol. 29, no. 3, pp. 244–256, Jul. 2020, doi: 10.3103/S1060992X20030029.

[17] S. Al and M. Dener, "STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment," *Comput. Secur.*, vol. 110, Nov. 2021, Art. no. 102435.

[18] C. A. de Souza, C. B. Westphall, R. B. Machado, J. B. M. Sobral, and G. D. S. Vieira, "Hybrid approach to intrusion detection in fog-based IoT environments," *Comput. Netw.*, vol. 180, Oct. 2020, Art. no. 107417, doi: 10.1016/j.comnet.2020.107417.

[19] B. Susilo and R. F. Sari, "Intrusion detection in IoT networks using deep learning algorithm," *Information*, vol. 11, no. 5, p. 279, May 2020, doi: 10.3390/info11050279.

[20] J. Liu, D. Yang, M. Lian, and M. Li, "Research on intrusion detection based on particle swarm optimization in IoT," *IEEE Access*, vol. 9, pp. 38254–38268, 2021, doi: 10.1109/ACCESS.2021.3063671.

[21] C. Tang, N. Luktarhan, and Y. Zhao, "An efficient intrusion detection method based on LightGBM and autoencoder," *Symmetry*, vol. 12, no. 9, p. 1458, Sep. 2020, doi: 10.3390/sym12091458.

[22] M. Alqahtani, A. Gumaei, H. Mathkour, and M. B. Ismail, "A genetic-based extreme gradient boosting model for detecting intrusions in wireless sensor networks," *Sensors*, vol. 19, no. 20, p. 4383, Oct. 2019.

[23] X. Tan, S. Su, Z. Huang, X. Guo, Z. Zuo, X. Sun, and L. Li, "Wireless sensor networks intrusion detection based on SMOTE and the random forest algorithm," *Sensors*, vol. 19, no. 1, p. 203, Jan. 2019, doi: 10.3390/s19010203.

[24] S. Ifzarne, H. Tabbaa, I. Hafidi, and N. Lamghari, "Anomaly detection using machine learning techniques in wireless sensor networks," *J. Phys., Conf. Ser.*, vol. 1743, no. 1, Jan. 2021, Art. no. 012021, doi: 10.1088/1742-6596/1743/1/012021.

[25] A. Yadav and A. Kumar, "Intrusion detection and prevention using RNN in WSN," in *Inventive Computation and Information Technologies*, vol. 336, S. Smys, V. E. Balas, and R. Palanisamy, Eds. Singapore: Springer, 2022, doi: 10.1007/978-981-16-6723-7_40.

[26] K. Sohraby, D. Minoli, and T. Znati, *Wireless Sensor Networks: Technology, Protocols, and Applications*. Hoboken, NJ, USA: Wiley, 2007.

[27] J. Kurose, V. Lesser, E. de Sousa e Silva, A. Jayasumana, and B. Liu, "Sensor networks seminar," Univ. Massachusetts, Amherst, MA, USA, Tech. Rep., CMPSCI 791L, Fall 2003.

[28] W. Dargie and C. Poellabauer, *Fundamentals of Wireless Sensor Networks: Theory and Practice*. Hoboken, NJ, USA: Wiley, 2010.

[29] Ž. Gavrić and D. Simić, "Overview of DOS attacks on wireless sensor networks and experimental results for simulation of interference attacks," *Ingeniería Investigación*, vol. 38, no. 1, pp. 130–138, Jan. 2018.

[30] D. Buch and D. C. Jinwala, "Denial of service attacks in wireless sensor networks," in *Proc. Int. Conf. Current Trends Technol.*, 2010, pp. 130–136.

[31] P. Schober, C. Boer, and L. A. Schwarte, "Correlation coefficients: Appropriate use and interpretation," *Anesthesia Analgesia*, vol. 126, no. 5, pp. 1763–1768, May 2018, doi: 10.1213/ANE.0000000000002864.

[32] X. Wang and F. Liu, "Data-driven relay selection for physical-layer security: A decision tree approach," *IEEE Access*, vol. 8, pp. 12105–12116, 2020, doi: 10.1109/ACCESS.2020.2965963.

[33] J. R. Quinlan, "Induction of decision trees," *Mach. Learn.* vol. 1, no. 1, pp. 10–81, 1986.

[34] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.

[35] R. Blagus and L. Lusa, "SMOTE for high-dimensional class-imbalanced data," *BMC Bioinf.*, vol. 14, no. 1, p. 106, Dec. 2013.

[36] P. Jeatrakul, K. W. Wong, and C. C. Fung, "Classification of imbalanced data by combining the complementary neural network and SMOTE algorithm," in *Proc. Int. Conf. Neural Inf. Process.*, Sydney, NSW, Australia, Nov. 2010, pp. 152–159.

[37] J. Wang, M. Xu, H. Wang, and J. Zhang, "Classification of imbalanced data by using the SMOTE algorithm and locally linear embedding," in *Proc. 8th Int. Conf. Signal Process.*, Beijing, China, Nov. 2006, pp. 16–20.

[38] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3146–3154.

[39] Z. Wen, J. Shi, B. He, J. Chen, K. Ramamohanarao, and Q. Li, "Exploiting GPUs for efficient gradient boosting decision tree training," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 12, pp. 2706–2717, Dec. 2019, doi: 10.1109/TPDS.2019.2920131.

[40] W. D. Fisher, "On grouping for maximum homogeneity," *J. Amer. Stat. Assoc.*, vol. 53, no. 284, pp. 789–798, Dec. 1958.

[41] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jun. 2016, pp. 785–794.

[42] S. Tyree, K. Q. Weinberger, K. Agrawal, and J. Paykin, "Parallel boosted regression trees for web search ranking," in *Proc. 20th Int. Conf. World Wide Web (WWW)*, 2011, pp. 387–396.

[43] J.-S. Pan, F. Fan, S.-C. Chu, H.-Q. Zhao, and G.-Y. Liu, "A lightweight intelligent intrusion detection model for wireless sensor networks," *Secur. Commun. Netw.*, vol. 2021, pp. 1–15, Apr. 2021.

[44] N. M. Zamry, A. Zainal, M. A. Rassam, E. H. Alkhammash, F. A. Ghaleb, and F. Saeed, "Lightweight anomaly detection scheme using incremental principal component analysis and support vector machine," *Sensors*, vol. 21, no. 23, p. 8017, Nov. 2021.

[45] H. Tabbaa, S. Ifzarne, and I. Hafidi, "An online ensemble learning model for detecting attacks in wireless sensor networks," 2022, *arXiv:2204.13814*.

[46] M. Mittal, R. P. de Prado, Y. Kawai, S. Nakajima, and J. E. Muñoz-Expósito, "Machine learning techniques for energy efficiency and anomaly detection in hybrid wireless sensor networks," *Energies*, vol. 14, no. 11, p. 3125, May 2021.

[47] D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011.

[48] N. Seliya, T. M. Khoshgoftaar, and J. Van Hulse, "A study on the relationships of classifier performance metrics," in *Proc. 21st IEEE Int. Conf. Tools Artif. Intell.*, Nov. 2009, pp. 59–66.

[49] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, no. 4, pp. 427–437, Jul. 2009.

[50] S. Yutaka, "The truth of the F-measure," Version 26, School Comput. Sci., Univ. Manchester MIB, Manchester, U.K., Tech. Rep., 2007. [Online]. Available: https://www.cs.odu.edu/~mukka/cs795sum09dm/Lecturenotes/Day3/F-measure-YS-26Oct07.pdf

[51] H. He and Y. Ma, "*Imbalanced Learning: Foundations, Algorithms, and Applications*, 1st ed. Hoboken, NJ, USA: Wiley, 2013.

MURAT DENER works as a Faculty Member of Gazi University. He is also the Head of the Information Security Engineering Department. He has been working in the field of Internet of Things, information security, and smart cities for nearly 15 years. He has more than 100 published international and national studies. He received the title of an Associate Professor in computer science and engineering.

SAMED AL is currently pursuing the Ph.D. degree with the Information Security Engineering Department. He has been working in the field of big data analytics, explainable artificial intelligence, and information security.

ABDULLAH ORMAN works as a Faculty Member of Yıldırım Beyazıt University. He has been working in the field of computer networks, database and data structures, artificial intelligence, and information security.