IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 20 June 2024, accepted 31 July 2024, date of publication 5 August 2024, date of current version 13 August 2024.

*Digital Object Identifier 10.1109/ACCESS.2024.3438619*

APPLIED RESEARCH

# A Novel Intrusion Detection Model for Enhancing Security in Smart City

**MAJED M. ABOROKBAH, (Member, IEEE)**
Faculty of Computers and Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia
e-mail: m.aborokbah@ut.edu.sa

**ABSTRACT** Preserving the smart city (SMC) ecosystem involves reducing environmental pollution and implementing effective waste management. Effective waste management is essential for minimizing pollution and ensuring a cleaner and more environmentally friendly atmosphere. Proper disposal of biodegradable materials plays a vital role in achieving this goal. Data gathering and utilization are key components of SMC initiatives. Nevertheless, there are apprehensions over the confidentiality and protection of data. SMC technologies are frequently linked to the internet, making them susceptible to attackers. To identify the attacks, a novel intrusion detection model called Smart Waste Management-Intrusion Detection System (SWM-IDS) is proposed in this research. The proposed model is developed using the Graph Neural Network (GNN) model and metaheuristic optimization algorithm. The key objective of this research is to classify and detect the attacks. The research model, SWM-IDS includes four main phases: data collection, data preprocessing, feature selection and classification. Initially, the CIC-IDS-2018 and CIC-DDoS-2019 datasets are collected to train and evaluate the research model. The data preprocessing phase includes data cleaning, unnecessary data removal, and normalization processes. After preprocessing, a binary variant of the Whale Optimizer (BWO) algorithm is used for selecting optimal features from the input dataset. Based on the selected features, the Graph SAmple and aggreGatE (GraphSAGE) model is implemented for classification. The SWM-IDS model is assessed in terms of detection rate, accuracy, f1-score, FPR, and precision. The model attained 99.72% accuracy, 99.69% detection rate, 99.65% f1-score, and 99.70% precision for the CIC-IDS-2018 dataset and attained 99.64% accuracy, 99.51% detection rate, 99.67% precision, and 99.58% f1-score for the CIC-DDoS dataset. These results were compared and validated with other models discussed in the literature review, and as compared, the research model outperformed all the other models.

**INDEX TERMS** IoT, smart waste management, GNN, binary WO algorithm, GraphSAGE.

## I. INTRODUCTION

Presently, the number of Internet of Things (IoT) linked devices exceeds 15 billion. It is projected that by 2030, the IoT will encompass over 29-billion devices, which is triple the times the number in 2020. The International Data Corporation predicts that IoT spending will increase significantly from 726-billion dollars in 2019 to 1.1-trillion dollars in 2023. IoT Analytics' worldwide IoT business expenditure dashboard predicts that the IoT enterprise market will see a compound annual growth rate (CAGR) of 19.4%, reaching a value of $483 billion between 2022 and 2027 [1]. Smart cities have greatly enhanced the life quality and services provided to inhabitants and urban landscapes. They have complete authority to manipulate physical things online and offer smart data to people on transportation, healthcare, smart infrastructure, public safety, parking management, garbage management, agriculture, and more [2]. Smart city programs have the capability to gather sensitive data. Nevertheless, distinct privacy and security concerns could emerge at various tiers of the architectural layers. Hence, it is crucial to know these privacy and security concerns during the process of building and implementing the applications [3]. By consolidating and analyzing all the data in an IoT

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

environment, the system becomes vulnerable to several risks. Furthermore, this system is highly vulnerable to significant malfunctions [4].

A smart city necessitates the implementation of secured transmission, monitoring, and feedback or response systems. Every component of a SMC is networked by a multitude of devices. These gadgets facilitate the connection between the residents of the SMC as well. Advanced urban infrastructure and cutting-edge technology enhance the physical structure and enhance the status of cybersecurity [5]. SMC infrastructure is susceptible to a range of cyber-attacks. A smart city utilizes a range of IoT devices and sensor networks to develop applications that are aware of its surroundings. The construction and architecture of sensor networks make them susceptible to cyber-attacks. As a result of the attacks on infrastructure, devices may become inaccessible, data may be lost, privacy breaches may occur for smart city people, and malignant software could be employed to spread false data and affect SMC infrastructures [6]. Disseminating inaccurate data could result in severe harm to the overall environment and lead to a considerable decline in living standards. Smart cities employ cutting-edge security systems to counteract any type of attack [7]. The IoT provides intelligent and self-configuring devices that are securely connected to global grid infrastructures, allowing for precise measurement and monitoring. The gadgets can provide improved security, performances, and dependability of the SMCs and their infrastructures. The fundamental structure of SMCs consists of four tiers: the interface, service, network, and perceptron, which are responsible for gathering sensitive data. Furthermore, it permits the network layers to facilitate the communication of two-way transmission, while the service layers examine sensitive information and the application layers offer the user a graphical user interface [8]. Figure 1 depicts the security issues and possible attacks present in IoT architectures.

A smart city's foundation lies in a smart environment, mostly used for addressing issues related to environmental degradation. Urban waste management is a regular activity that demands a substantial number of labour and affects economic, social, efficiency, and environmental factors [9]. Efficient waste management significantly affects the overall well-being of the population. Sensors, GPS, and LED technology may be employed to effectively control and monitor waste in garbage cans. The sensor autonomously alerts the operator of the optimal time to get the garbage can and provides the most efficient path, disregarding traffic conditions. Once the garbage can is full, a notification will be transmitted to the administrator, who later communicates the position to the garbage collector provided with a smartphone. Subsequently, the collector proceeds to the designated location, collects trash, and changes the container [10].

### A. PROBLEM STATEMENT

The IoT has lately transformed into a cutting-edge technology used to construct smart environments, specifically smart waste management. Security and privacy are crucial concerns in any practical smart environment that operates on the IoT framework. The security vulnerabilities present in IoT systems give rise to security concerns that impact applications designed for smart environments [11]. Therefore, it is imperative to develop IDSs specifically tailored for IoT-based SWM settings to counteract security threats targeting IoT devices by exploiting their weaknesses. Regular and common IDSs may not be appropriate for IoT platforms because of the restricted computation and storage capabilities of IoT devices and the usage of unique protocols [12]. Hence, this research proposes a novel IDS model for identifying attacks using the GNN method. This research highlights a binary classification issue for intrusion classification wherein all the observations are classified as an attack or normal class.

### B. RESEARCH CONTRIBUTION

In this research, a hybrid model called SWM-IDS is proposed, which used the DL and optimization techniques to identify attacks. The research model includes data collection, data preprocessing, feature selection, and classification processes. The CIC-DDoS and CIC-IDS data sets are used to train and evaluate the research model. In data preprocessing, data cleaning, unnecessary data removal, and normalization processes are performed. After preprocessing, the feature selection is carried out using the BWO algorithm. Based on the selected attributes, the GraphSAGE model was used to detect and classify the attacks. The proposed SWM-IDS model's performance is evaluated based on accuracy, detection rate, precision, FPR, and f1-score. Based on the research contribution, the following presents the research objectives:

- To develop a novel intrusion detection model called SWM-IDS designed for identifying attacks.
- To utilize the CIC-DDoS and CIC-IDS datasets for training and evaluating the performance of the proposed SDN-IDS model in identifying attacks.
- To apply the BWO algorithm for feature selection to choose the most significant features for identifying attacks within the smart IoT environment.
- To employ GraphSAGE, a GNN model to perform identification and classification of attacks based on the selected features.
- To evaluate the performance of the proposed SWM-IDS model using key metrics such as accuracy, detection rate, precision, FPR, and f1-score to assess its effectiveness in detecting and classifying DDoS attacks accurately and efficiently.

The subsequent sections of the research are structured as follows: Section II discusses the investigation of the existing attack detection models. Section III provides the presentation of the materials and methods employed in this research. The implementation of the proposed SWM-IDS model is presented in 4th section. Section V includes the discussion of experimental analysis and the comparison of findings. In the

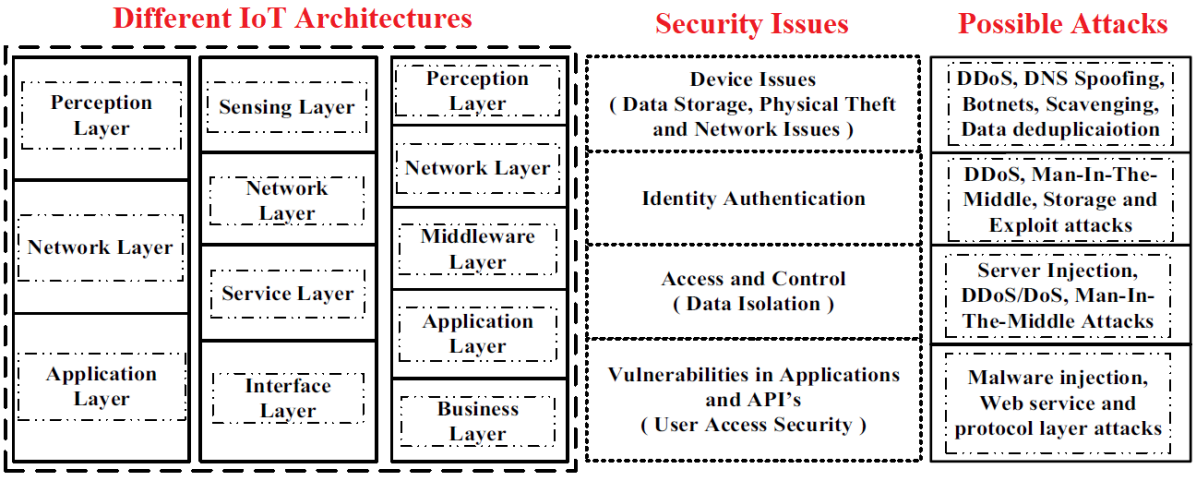


**FIGURE 1.** Attacks and issues in IoT-Based smart city.

final section, the research concludes the work and suggests the possible areas for further research.

## II. LITERATURE REVIEW

The current SMC infrastructure heavily relies on IoT technology. Nevertheless, the security concerns linked to IoT network exposes SMC architectures to cyber-attacks. The DDoS attack breaches the authorization requirements in SMC infrastructure. The research [13] presented the development of a hybrid DL model called Convolutional Neural Networks-Restricted Boltzmann Machines (CNN-RBM) for the purpose of identifying DDoS and replay attacks in a real-time SMC architecture. The model's performance was assessed by artificially producing replay and DDoS attack data. The findings demonstrated a notable improvement to the model by the incorporation of the RBM. Real-time detection of intrusion on IoT devices is crucial for ensuring the reliability, security, and profitability of IoT-enabled services. An intrusion detection solution for IoT devices based on deep neural network was proposed [14]. This solution utilized a 4-layered Fully Connected (FC) deep neural network (DNN) architecture to identify malevolent network activity that might potentially launch attacks on interconnected IoT devices. The created system was designed to be independent of communication protocols to simplify the deployment process. The model exhibited consistent and dependable performance when tested with both simulated and actual attacks in the empirical performance analysis.

Various strategies are available to defend DDoS attacks in a SMC, however, several difficulties persist. The study [15] introduced a robust solution designed to counteract DDoS attacks in smart cities. This work utilized machine learning (ML) models and SDN security controllers with optimization techniques to mitigate the effects of typical DDoS intrusions on SMCs. The study employed a software-defined network (SDN) that relied on a detection system based on a ML approach and security controllers. The XGBoost model successfully classified network traffic with minimal false positives. An IDS defensive model that enhanced the security of IoT networks by employing anomaly detection and ML techniques was proposed to counteract DoS attacks [16]. The IDS model utilized anomaly detection to constantly monitor network traffics for any variations from usual patterns. To achieve this objective, four distinct supervised classifier methods were employed: Support Vector Machines (SVM), Decision Trees (DT), K-Nearest Neighbours (kNN), and Random Forests (RF). Furthermore, two distinct feature selection techniques, Correlation-based Features Selection (CFS) and Genetic Algorithm (GA) were employed. The DT and RF classifiers had the highest scores when trained with characteristics selected by GA.

The study [17] focused on performing anomaly identification in IoT devices functioning in SMC environments, for protecting against cybersecurity attacks. An extensive analysis was conducted on a variety of ML approaches, including both conventional algorithms such as Naive Bayes, RF, SVM, and neural networks, as well as advanced methodologies like split learning (SL) and federated learning (FL). Through testing with the datasets, useful aspects into the advantages and drawbacks of different techniques were obtained. FL and SL were effective solutions for achieving a balance between data privacy and detection accuracy. A billiard-based optimizer with DL-enabled anomaly detection and classification method was proposed in [18] for IoT-assisted sustainable SMCs. The correct identification and categorization of abnormalities in the SMC supported by the IoT was the aim of this approach. The binary pigeon optimizer method was used by the method to achieve that for the effective feature selection. The Elman recurrent neural network approach was used for anomaly identification and categorization. The outcomes showed the potential performance of the model.

The study introduced a DL architecture that utilized a dense random neural network technique to differentiate and categorize abnormal behaviors from regular ones, based on the specific sort of attack on the IoT [19]. ML algorithms

have a limited capacity to examine performance when compared to DL models. Notably, the analysis of DL neural network structures achieved improved computational efficiency and produced satisfactory outcomes for categorical attacks. The study [20] presented an extensive literature analysis that covered many cybersecurity concerns, with a specific focus on DoS assaults. The study emphasized that DoS assaults provided a substantial difficulty to both the IoT and Information-Centric Networking (ICN). To improve security measures against these attacks, ML approaches, including SVM, RF, and KNN, were employed for the detection and mitigation of such attacks.

Feature selection recognizes a subset of IoT-related attributes that describe traffic aspects and separate malicious from benign activity for IoT with DDoS attack detection. The work [21] developed an IoT-based snake optimizer (SO) with ensemble learning DDoS attack detection method. The method detected DDoS attacks automatically. The method selected feature subsets using the SO algorithm. Additionally, an ensemble of bidirectional long short-term memory (BiLSTM), LSTM, and deep belief networks (DBN) were used. The Adadelta optimization tuned DL algorithm parameters. Results showed that the method provided better performance. The study [22] examined smart home cyberattacks. The authors reverse-engineered IoT firmware images to extract names and passwords, determined CPU architecture and launched DoS intrusions to malfunction the devices. The researchers utilized the UPnP service for scanning and divulging critical device data, which was utilized to integrate rules or ports to transform IoT devices into routers to attack various devices. The IoT network was linked to an intrusion detection and prevention system (IDPS) for mitigating the attacks. The IDPS model detected attacks better.

A chaotic poor and rich optimizer with DL methodology (IDCPRO-DLM) was proposed in [23] for pervasive and smart network intrusion detection. The model selected feature subsets using a CPRO algorithm. A butterfly optimizer approach (BOA) with deep sparse auto encoder (DSAE) was employed to detect attacks. Evaluation on the benchmark CICIDS dataset shows higher performance of the model with an average accuracy of 98.53%. IDS detects network irregularities and takes action to keep IoT applications secure and reliable. The Deep migration learning model and intrusion detection technology were used in [24] to develop a smart city IoT features extraction and IDS method. Migration learning and data features extraction were applied in this work. The results indicated that the model has increased detection rate, decreased false positives, and better efficiency.

An IDS model was developed to in [25] examine the acquired network traffic information inside a big data framework and identified network threats using a DL algorithm. The features in the dataset were diminished through the utilization of the correlation approach, guaranteeing the integration of best attributes in the evaluation. The hybrid DL method was created by combining a one-dimensional CNN with a LSTM model. The hybrid DL method in this study achieved the best level of accuracy. DDoS attacks are increasing in IoT, requiring stronger protection and authentication. Recurrent Neural Network (RNN)-based IoT network threat mitigation model was proposed in [26]. RNN classified attack characteristics using pre-processed and feature-extracted data. After preprocessing datasets using min-max scaling, XGBoost choose the features. In testing and training data sets, the RNN-based architecture achieved good classification accuracy. The study [27] investigated the concept of a SMCs security IDS to enhance the performance of a conventional IDS for IoT applications in SMCs. This study presented the development of a Smart Attacks Learning Machine Advisor (SALMA) for SDN in smart cities. SALMA was a multilayer hybrid IDS based on Extreme Learning Machine (ELM) to safeguard SMCs from various smart intrusions. The model relied exclusively on six flow characteristics utilized in SDNs. The system offered a highly efficient technique for identifying intrusions in SDNs.

## III. RESEARCH GAP

A review of existing research in IoT intrusion detection for smart city infrastructures has implemented various approaches. These include hybrid DL models, ML-based anomaly detection, and optimization techniques. The studies addressed numerous aspects of IoT security, such as DDoS attack detection, anomaly detection, and smart city threat mitigation. However, despite the progress made in this area, several research gaps remain to be addressed.

- First, while many existing models target specific attack types like DDoS or DoS, a need exists for comprehensive IDS capable of detecting and mitigating a wider range of cyber threats targeting IoT devices in smart city environments.
- Second, most studies primarily evaluate their proposed models using simulated or artificially generated attack data. This data may not fully capture the complexities and variations of real-world attack scenarios. Consequently, there is a lack of empirical validation for these models in real-world smart city environments.
- Furthermore, the scalability and deployment challenges of these IDS in large-scale IoT networks require further investigation and solutions. Additionally, research focusing on improving the explainability and interpretability of intrusion detection models is necessary to facilitate trust and adoption by stakeholders in smart city infrastructures.
- Finally, considering the dynamic nature of IoT networks and evolving cyber threats, there is a need for adaptive and self-learning IDS. These systems should continuously adapt to emerging threats and evolving network conditions.

Therefore, the proposed research aims to address these gaps by developing a robust and scalable IDS. This system will be based on Graph Neural Networks and specifically developed for smart waste management platforms. It will provide comprehensive protection against various cyber

threats while ensuring real-time detection and adaptability to dynamic network environments.

## IV. MATERIALS AND METHODS

This section provides a brief introduction to the methods and materials used for the development of the proposed SWM-IDS model. As the proposed SWM-IDS model includes data collection, preprocessing, feature selection and classification processes, this section provides the details of the datasets used, data preprocessing methods, feature selection method and classification model. Figure 2 depicts the architecture of the architecture of proposed SWM system.

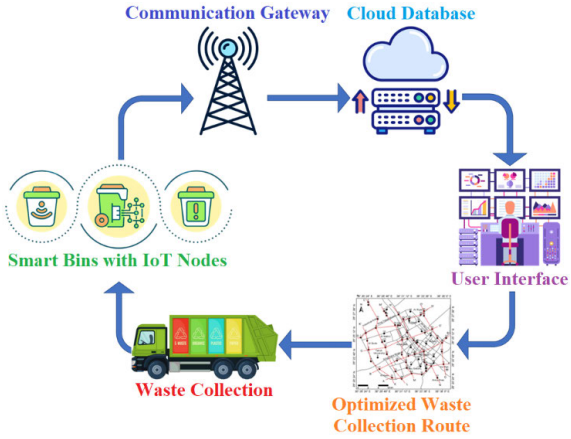


**FIGURE 2.** Proposed architecture of IoT-Based SWM system.

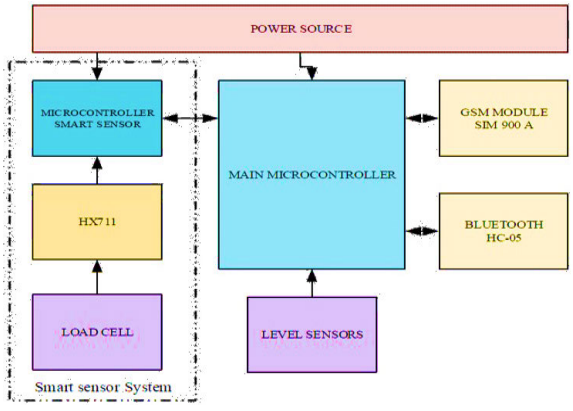


**FIGURE 3.** Hardware model of SWM system.

Analyzing data provided by IoT devices in garbage bins, collection trucks, and landfill sensors can help discover trends and patterns in waste creation, efficiency of collection, and rate of recycling. By utilizing a data-driven strategy, waste management firms and municipalities may enhance the efficiency of their operations and make well-informed decisions. The SWM system can be integrated into standard garbage bins. It comprises sensing units, a GSM module and Bluetooth for data transmission, as well as a web-based monitoring system and mobile application for communicating and interfacing with the waste department for efficient SWM. The overall structure of the SWM is represented in Figure 3.

GSM modules connect with servers. It sends trash weight and bin capacity data regularly. When the garbage container is full, it alerts workers to pick it up. Bluetooth is hooked for short-range communication. Workers utilize it for system maintenance. It also gets data from the application if the GSM module fails. Bluetooth connects to a smartphone app to share waste-bin weight. Smartphone software helps workers choose and monitor garbage bins for effective waste management. GSM module notifications appeared in the mobile app. It can speed up a full waste-bin handle. Web-based monitoring manages all waste-bin data. Daily, weekly, monthly, and annual graphics will display citywide trash. The website is developed with a code igniter and the database with PHP and MySQL.

### A. DESCRIPTION OF DATASETS

The dataset is the most essential part of the proposed SWM-IDS model, which is directly related to the processing and efficiency of the model. In this research, the CIC-DDoS-2019 and CIC-IDS-2018 data sets are used to train and evaluate the research model. The CIC-IDS data set comprises a flow of network characteristics with 80 attributes. These features encompass seven distinct attack conditions, namely Web Attack, Heartbleed, DoS, Internal Penetration, DDoS, Brute Force, and Botnet [28]. The CIC-DDoS2019 data set was currently the most extensive and recent collection of DDoS intrusion data that was publicly accessible. It includes 88-dimensional network flow features such as protocol, destination port, and timestamp. The dataset encompasses 13 distinct types of DDoS attacks, including PORTMAP, LDAP, MSSQL, NTP, SSDP, NetBIOS, CharGen, UDP Flood, DNS, UDP Lag, TFTP, Syn Flood, and SNMP [29].

### B. DATA PREPROCESSING

Preprocessing the data is crucial to deep learning because it can boost training efficiency. Irrelevant data removal, missing values/outlier detection, label conversion, and normalization are necessary steps in preprocessing for this research. Removing irrelevant data includes features such as "Unnamed: 0," "Flow ID," and "Inbound," which are not useful for this research. IP socket or feature ID-related features like "Timestamp" IP address and port number information may overfit the model [30]. Therefore, unnecessary features were eliminated and only the necessary ones were preserved. Next preprocessing is handling missing values after detecting significant features. Several methods can deal with missing values. Because the median of the feature better represents the majority value, the proposed approach substitutes missing data with it. When data is distorted or has outliers, the median estimates well. To calculate the median of a set of values, equations (1) and (2) were used [31].

$$\left(\frac{n+1}{2}\right)^{th}, \quad for\ odd\ n \tag{1}$$

$$\left(\frac{(n/2)+\left((n+2)/2\right)}{2}\right)^{th}, \quad for\ even\ n \tag{2}$$

**TABLE 1.** Comparative analysis of reviewed studies.

| Approach | Application Scenario | Advantages | Disadvantages |
|---|---|---|---|
| Hybrid DL model: CNN-RBM [13] | Real-time smart city platform, detecting replay and DDoS attacks. | Improvement with RBM incorporation, and real-time detection. | Artificially produced attack data, computational complexity. |
| Deep Fully Connected NN [14] | Intrusion detection on IoT devices, simulated and actual attacks. | Independent of communication protocols, consistent performance. | Limited to FC architecture, and scalability issues. |
| ML models with SDN [15] | Mitigating DDoS attacks in smart cities. | Utilization of SDN and ML, minimal false positives. | Dependency on SDN infrastructure, deployment complexity. |
| Anomaly detection with ML [16] | Enhancing the security of IoT networks against DoS attacks. | Utilizes multiple ML classifiers and feature selection methods. | Limited to supervised learning, and scalability concerns. |
| ML approaches, federated learning [17] | Detecting anomalies in IoT devices in SMC environments. | Effective balance between data privacy and detection accuracy. | Complexities of federated learning, computational overhead. |
| DL-enabled anomaly detection [18] | Anomaly identification and categorization in IoT-assisted cities. | Utilizes DL and optimization methods, potential performance. | Complexity of DL models, computational resources. |
| DL architecture [19] | Differentiation of abnormal behaviors in IoT. | Improved computational efficiency, and satisfactory outcomes. | Complexity of DL models, interpretability issues. |
| ML approaches [20] | Detection and mitigation of DoS attacks in ICN and IoT. | Utilizes various ML algorithms, detection, and mitigation. | Generalization to other attack types, real-world validation. |
| Ensemble learning with SO [21] | DDoS attack detection in IoT. | Automatic feature selection, improved performance. | Complexity of ensemble methods, computational resources. |
| Reverse engineering, IDPS [22] | Examination of smart home cyberattacks. | Detection of attacks through IDPS, empirical analysis. | Dependence on firmware vulnerabilities, ethical concerns. |
| Chaotic optimizer with DL [23] | Intrusion detection in pervasive and smart networks. | Utilizes DL and optimization methods, high performance. | Complexity of optimizer algorithms, computational resources. |
| Deep migration learning model [24] | Feature extraction and intrusion detection in smart city IoT. | Increased detection rate, decreased false positives. | Interpretability issues, computational complexity. |
| Hybrid DL method [25] | Intrusion detection inside a big data framework. | Hybrid DL method, effective feature selection. | Computational complexity, data preprocessing requirements. |
| RNN-based IoT network model [26] | Mitigation of DDoS attacks in IoT. | Good classification accuracy, preprocessed and feature-extracted. | Limited to recurrent architecture, and computational complexity. |
| Hybrid IDS with SDN [27] | Enhancing performance of IDS for IoT applications in smart cities. | Multilayer hybrid IDS is an efficient technique for identifying intrusions. | Dependency on SDN infrastructure, scalability concerns. |

**TABLE 2.** Distribution of datasets.

| Data | Malicious | Benign | Total | |
|---|---|---|---|---|
| Training | 2195618 | 10787766 | 12983384 | CIC-IDS-2018 |
| Test | 549647 | 2696942 | 3246589 | |
| Training | 50006249 | 56863 | 50063112 | CIC-DDoS-2019 |
| Test | 20307560 | 56965 | 20364525 | |

Here $n$ represents the total dataset samples. This research examines the process of classifying network traffic into two distinct classes: normal and attack. Mathematically, this function can be expressed as follows.

$$f(l) = \begin{cases} Attack, & for\ l \neq Normal \\ Normal, & for\ l = Normal \end{cases} \quad (3)$$

Here $l$ represents the label of data. The dataset exhibits a wide range of continuous values, which results in increased prediction errors. Normalization of the dataset greatly reduces classification errors and enables the model to converge more quickly. This work utilized Min-Max scaling to normalize all features in the dataset, considering the varying scales present. Min-Max scaling is a linear transformation applied to the original data to achieve normalization. The scaling formula is defined by the following equation, where $d$ reflects the original data and $d'$ reflects the transformed data:

$$d' = \frac{(d - min)}{(max - min)} \quad (4)$$

Within this context, the terms *max* and *min* denote the maximum and minimum values seen in the column where the parameter $d$ is located. This normalization technique has a broad scope of applicability. By utilizing this methodology, the data is converted to a range of values between 0 and 1, while maintaining the original data structures. This distinguishes it from the Z-Score method. As a result, this Min-Max approach enables fast and simple data normalization within a given range of values [32].

### C. BINARY WO ALGORITHM

The WO algorithm is simplistic and efficient in its ability to investigate global solutions. A multitude of researchers have been drawn to the task of identifying a decrease inside an information system. An elementary approach involves identifying a group, generating every potential group, and selecting the groups with the lowest value of function. This technique was feasible only for a simple dataset. In general, a single group of characteristics was utilized to minimize a data set, thus the computations required for identifying the others are usually unnecessary.

In the context of the feature selection issue, a solution refers to a binary vector that precisely represents the current subset of features. The perturbation of a subset occurs when a small number of individuals inside the subset are randomly changed, resulting in the activation of a specific form of the WO algorithm. The objective functions might be a rough set that quantifies whether the solution contains redundant features. It can also consider criteria such as classification accuracy or other factors that balance the computational load of attribute extraction with efficiency [33]. In this part, the binary version of the WO algorithm (BWO) was employed to perform feature selection using both rough set and wrapper techniques, which will be further elaborated on. Figure 4 represents the workflow of the research model.

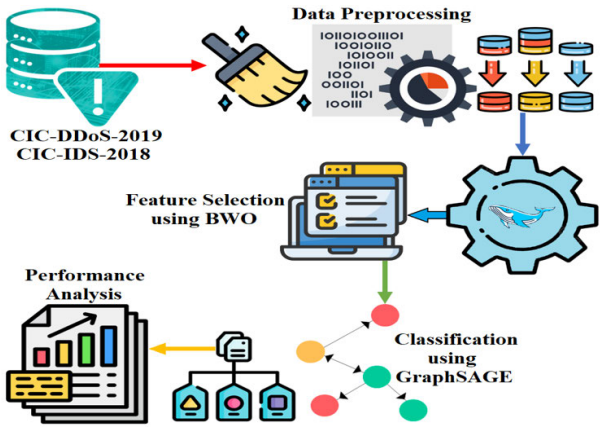


**FIGURE 4.** Workflow of the proposed model.

### D. GRAPH NEURAL NETWORK

It is a new field of ML that shows great potential in addressing practical challenges, including recommender systems, social networks, pattern recognition, and computer vision. A key component of GNN is the operators, which were reliable for training GNN on data that was graph-structured and transmitting information about learning nodes to various layers. GNN is a rapidly expanding subfield within the field of ML. The power and promise of this technology reside in its capacity to use an underlying graph format of large amounts of information seen in many practical application areas. The graph structure represents structural data by depicting a collection of items and their interconnections. Graph nodes represent the things, whereas graph edges reflect their relationships. In a network, individual hosts (identified by their IP addresses) were represented as the nodes in a graph, while the communication between these hosts, known as network flows, was represented as the edges in the graph. The primary reason for utilizing a GNN in intrusion detection is its capacity to effectively and immediately leverage the extensive structural data included in the flow data of the network, which could be readily represented in a graph structure. GraphSAGE is well-suited for learning representations of nodes in graph-structured data, making it a suitable choice for analyzing network traffic data and detecting intrusions [34].

## V. PROPOSED BWO-GNN MODEL

This section presents the implementation of the proposed model regarding feature selection and classification. Figure 5 represents the architecture of the proposed BWO-GNN for SWM-IDS.

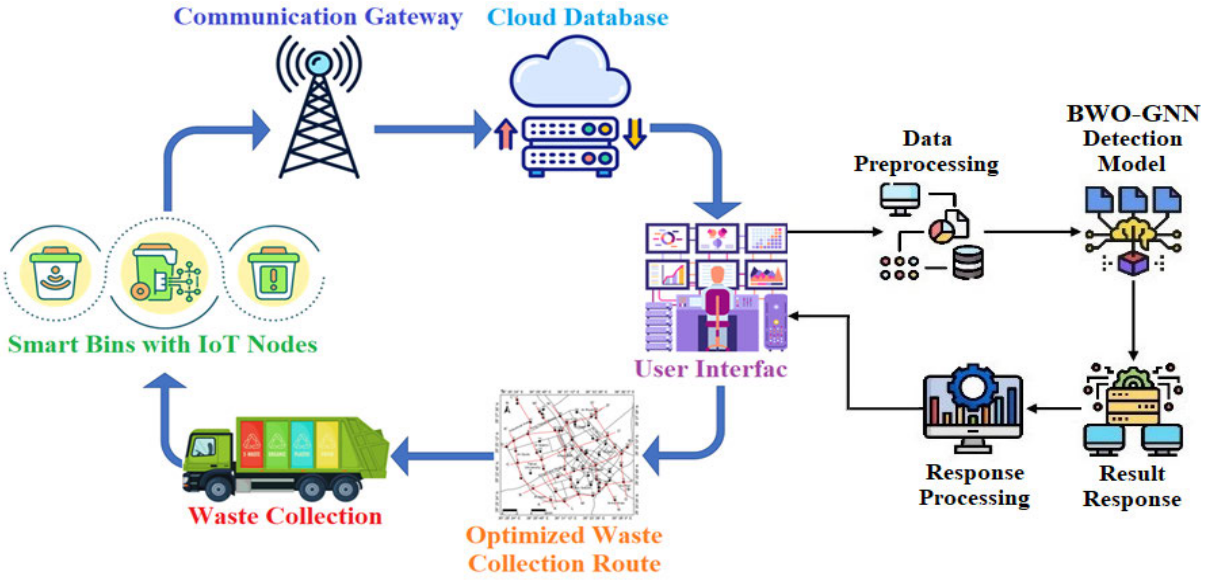


**FIGURE 5.** Architecture of the BWO-GNN model for SWM-IDS.

### A. FEATURE SELECTION USING BWO

The WO algorithm employs a group of search operators to ascertain an optimal global value for an optimization issue. The searching initiates by postulating a collection of stochastic solutions for a specified issue. Next, the operators will attempt to adjust its placements over the most optimal search operator till they reach a specified termination condition. Humpback whales perform in a synchronous spiral-shaped movement while swimming around their prey inside a diminishing circle. Therefore, to quantitatively represent this simultaneous behavior, we assume a chance of 0.5 to determine whether to update the spiral or shrinking encircling mechanism throughout the optimization process. The WO representation can be described as represented in equation (5):

$$\vec{X}(t+1) = \begin{cases} \vec{\hat{X}}(t) - \vec{A} \cdot \vec{D}, & if\ p < 0.5 \\ \vec{D} \cdot e^{bl} \cdot \cos(2\pi l) + \vec{\hat{X}}(t), & if\ p \geq 0.5 \end{cases} \tag{5}$$

The first section of equation (5) models the process of encircling, whereas the subsequent component replicates the spiral update of location. Let $p$ be a random integer between 0 and 1, $t$ represents the present iteration, $\hat{X}$ be the best solution's position vector achieved thus far, $b$ be a constant used to define the form of a logarithmic spiral, and $l$ be a random value between $-1$ and 1. $\vec{D}$ is determined as follows.

$$\vec{D} = \left| \vec{C} \cdot \vec{\hat{X}}(t) - \vec{X}(t) \right| \tag{6}$$

whereas, $\vec{C}$ and $\vec{A}$ are coefficient vectors, determined using the following calculations:

$$\vec{A} = 2\vec{a} \cdot \vec{r} - \vec{a} \tag{7}$$

$$\vec{C} = 2 \cdot \vec{r} \tag{8}$$

where the value of $\vec{a}$ linearly decreases from two to zero over the duration of iterations, and $\vec{r}$ was a random vector within the range of [0, 1].

The distance between the prey and ith whale, which represents the best optimal solutions achieved thus far, was represented as:

$$\vec{\hat{D}} = \left| \vec{\hat{X}}(t) - \vec{X}(t) \right| \tag{9}$$

The WO algorithm achieves a favorable equilibrium among the exploration step, which involves searching for food, and the exploitation step, which involves Bubble nets attacking. It seamlessly transitions among both stages by dynamically reducing a search vector 'A.' Few iterations were dedicated for exploration, with the optimal solution serving as the key point for updating the location of additional search operators if $|A| \geq 1$. During the remaining iteration cycles, the focus is on exploiting the best solution found thus far, which serves as the pivotal moment when $|A| < 1$.

The objective of the BWO is to address binary optimization difficulties. During the whale optimization process, search agents constantly adjust their placements to the most optimal spot inside the search region, based on the best optimal search operator found. However, in the BWO algorithm, every solution was located at the vertices of a hypercube. Additionally, the group of solutions was represented in a binary format at all iterations. The search operators in the algorithm change their locations based on Eq (5) while maintaining a binary constraint. The continuously upgraded location was compressed utilizing the sigmoid functions, and hence the resulting values are randomly threshold to get the binary positions updated. A binary location $\vec{X}_d^{t+1}$ in $d$

dimension at $t$ iteration was determined by Eq (10) as a result.

$$\vec{X}_d^{t+1} = \begin{cases} 1, & if\ sigmoid\left(\vec{X}^t\right) \geq rand \\ 0, & otherwise \end{cases} \tag{10}$$

In this context, *rand* represents a randomly generated integer that follows a uniform distribution between 0 and 1. The sigmoid function, denoted as $sigmoid(a)$, is defined as:

$$sigmoid\ (a) = \frac{1}{1 + e^{-10(a-0.5)}} \tag{11}$$

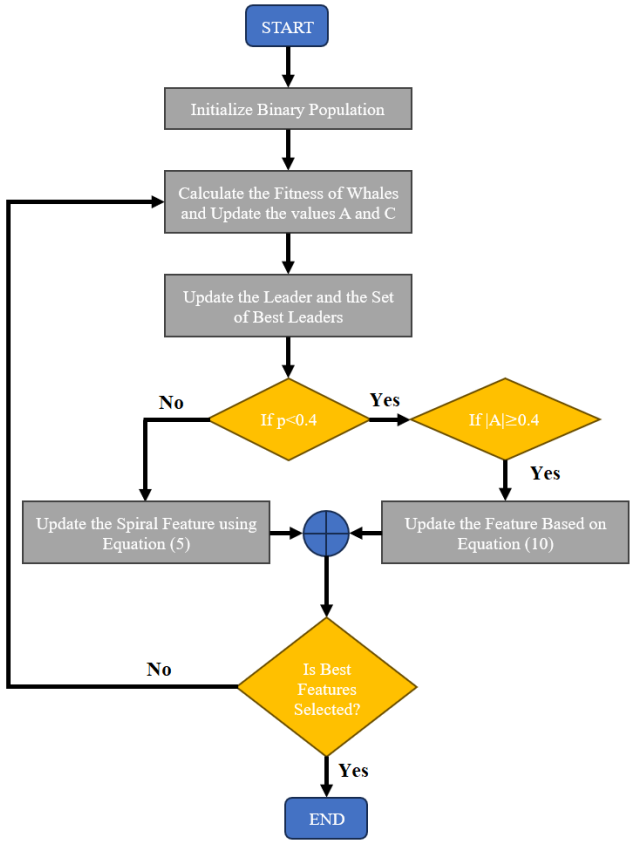


**FIGURE 6.** Architecture of GraphSAGE.

The binary problem of feature selection is represented by an n-dimensional Boolean matrix as the search space. Therefore, the BWO method is suitable for solving feature selection issues. Given the task of choosing whether to choose a certain attribute, the whale's location was denoted by binary vectors. A value '1' signifies the selection of the feature, whereas a value '0' signifies the non-selection of the feature. The primary objectives of feature selection are to optimize classification accuracy and minimize the number of features used. The BWO algorithm is employed to dynamically explore the optimal combination of features, considering both objectives. The BWO uses a fitness function to assess the placements of individual whales. Figure 6 represent the flowchart of the BWO.

$$FF = \alpha E_R + (1 - \alpha)\frac{|S^*|}{|S|} \tag{12}$$

The variables $E_R$ represents the error rate of classifications of the features selected, $S^*$, which represents the total features selected, and $S$, which represents the overall features. The symbol $\alpha$ and $(1 - \alpha)$ denote the significance of the classifier's accuracy and the quantity of features chosen, respectively, within the range of $\alpha \in [0.5, 1]$ [35].

Using this BWO technique, Destination Port, Protocol, Total Length of Bwd Packets, Fwd Packet Length Max, Fwd Packet Length Mean, Fwd Packet Length Std, Flow Bytes/s, Flow IAT Max, Fwd IAT Max, Fwd Header Length, Bwd Header Length, Max Packet Length, Packet Length Mean, Packet Length Variance, ACK Flag Count, Average Packet Size, Avg Fwd Segment Size, Subflow Bwd Bytes, Init_Win_bytes_forward, Init_Win_bytes_backward, and min_seg_size_forward features were selected and used for classification.

### B. CLASSIFICATION USING GRAPHSAGE

The GraphSAGE method is well recognized as one of the most prominent GNNs. GraphSAGE randomly selects a subset of a predetermined size by uniform sampling. This enables the technique to restrict the time and space complexities, regardless of the graph structure (such as the distribution of node degrees) and the size of the batch. The GraphSAGE method functions on a graph $G\,(V\,,\varepsilon)$, in which $V$ represents the nodes collection and $\varepsilon$ represents the edges collection. The characteristics of a node $v$ were denoted by a vector $x_v$, and the feature vectors in the entire collection of nodes were represented as $\{x_v,\ \forall v \in V\}$. An essential hyperparameter in the GraphSAGE method is the value of K, which determines the total convolutional layers of graph. This value indicates the total iterations through which node data was collected. Another crucial feature of GraphSAGE was the selection of a function called differentiable aggregator $AG_k,\ \forall k \in \{1, \ldots K\}$, which was used to combine data from neighboring nodes. The method was executed through both a backward and forward propagation phase, as thoroughly explained following.

Node Embedding: Like the convolutional process in CNNs, the node embedding is computed by collecting information about the node's local neighborhood. The GraphSAGE method begins with the assumption that the model has already undergone training and that the aggregator function and weight matrices parameters remain unchanged. The method systematically collects and combines data from the node's immediate neighbors, as well as the neighbors of its neighbors, and so on. During each iteration, the node's neighborhood is first sampled, and the data from the nodes sampled are then combined into a vector. Equation 13 expresses the aggregated data $h^k_{N(v)}$ at a node $v$ in the k-th layer, which is based on the sampled neighborhood $N(v)$.

$$h^k_{N(v)} = AG_k\left(\left\{h^{k-1}_u,\ \forall u \in N\,(v)\right\}\right) \tag{13}$$

In this context, the term $h^{k-1}_u$ denotes the representation of a node $u$ in the preceding layer. Every node's embedding $u$ in

the vicinity of $v$ was combined to form the node embedding $v$ at the $k$ layer. The task of aggregating the node and topological properties in the graph, gathered, and combined from every graph node's k-hop neighborhood, is depicted in Figure 7. The GraphSAGE model provides many aggregation techniques, such as mean, pooling, or alternative neural network architectures like Long Short-Term Memory (LSTM).

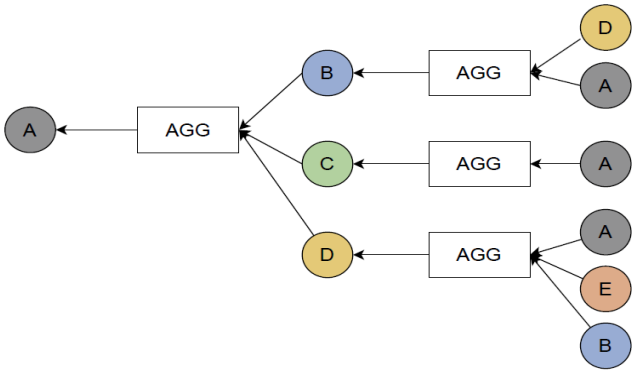


**FIGURE 7.** Architecture of GraphSAGE.

The embeddings combined of the selected neighborhoods $h^k_{N(v)}$ were subsequently merged with the embedding of node from the preceding layers $h^{k-1}_u$. A calculation of the embedding for node $v$ in $k$ layer was done by applying the trainable weight matrix $W^k$ of the model and transferring the output through a function of non-linear activations $\sigma$ (ReLU), as indicated in Equation 14.

$$h^k_v = \sigma\left(W^k \cdot CONCAT\left(h^{k-1}_u, h^k_{N(v)}\right)\right) \tag{14}$$

The node $v$ was represented by the embedding $z_v$ at the final layer $K$, as indicated by Equation 15 [36].

$$z_v = h^K_v, \ \forall v \in V \tag{15}$$

To classify nodes, the value $z_v$ can be processed using either a softmax layer or a sigmoidal neuron. The primary reason for utilizing a GraphSAGE in IDSs is its capacity to effortlessly and immediately leverage the abundant structural data present in the data's network flow, which could be readily represented in the graph structure. The model is a generic inductive approach that utilizes node feature data to effectively produce node embeddings for data that has not been seen before.

## VI. EXPERIMENTATION ANALYSIS

### A. EXPERIMENTAL SETUP

This section provides an overview of the evaluation and performance analysis conducted on the proposed SWM-IDS model. The research model was tested in an experimental configuration using a PC with Windows 10, a 64-bit operating system, an Intel(R) i7 processor running at 4.60 GHz, and 16GB of RAM. The model was developed using the Python 3.11.4 64-bit tool and relied on the Pandas, Numpy, and Scikit-learn modules. The achieved outcomes for the SWM-IDS model are compared and verified against the existing IDS models.

### B. PERFORMANCE METRICS

The evaluation of the research model was conducted based on parameters including accuracy, detection rate (recall), false positive rate (FPR), f1-score, and precision. The metrics are calculated using the values of True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN), which are the four categories utilized to identify the results of the detection. TP is the count of attack samples that were accurately identified as attack samples by the model. The count of normal samples accurately identified as normal samples was designated as TN. FP is the count of normal samples that were incorrectly classified as attack samples by the model. The FN value represents the number of attack occurrences that the model incorrectly classified as normal samples.

Accuracy, in the context of intrusion detection, refers to the ratio of rightly identified outcomes to the total samples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{16}$$

Precision can be measured by the ratio of true positive samples to the total samples recognized as positives, specifically focusing on the correct detection of negative samples as positive.

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

The detection rate refers to the ratio of positive samples to the total positive samples.

$$DetectionRate = \frac{TP}{TP + FN} \tag{18}$$

The F1 score is a statistical metric used to assess the accuracy of positive class detections. The f1-score is the ratio of correctly identified positive samples to all samples identified as positive.

$$F1score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{19}$$

FPR in an IDS stands for False Positive Rate, which is the proportion of alerts produced by the system that are inaccurate or non-malicious occurrences.

$$FPR = \frac{FP}{FP + TN} \tag{20}$$

False positives can arise due to a range of circumstances, including misconfigurations, obsolete rules or signatures, or regular traffic that activates the IDS. Hence, it is crucial to prioritize the reduction of false positives while designing and implementing an IDS.

### C. PERFORMANCE EVALUATION

This section provides an in-depth examination of the findings of the SWM-IDS model. An evaluation of the research model's ability to detect intrusions was carried out by assessing its performance using performance measures. Table 3 presents the results of the SWM-IDS model on the CIC-IDS-2018 data set. The SWM-IDS model achieved

**TABLE 3.** Results of the SWM-IDS model on CIC-IDS dataset.

| Metric | Training Results | Testing Results |
|---|---|---|
| Accuracy | 99.84 | 99.72 |
| Detection Rate | 99.79 | 99.67 |
| Precision | 99.85 | 99.70 |
| F1-score | 99.80 | 99.65 |
| FPR | 0.32 | 0.41 |

an accuracy of 99.84% during training, demonstrating that it properly classified the instances in the training dataset. During testing, the model maintained a high accuracy of 99.72%, demonstrating its ability to generalize well to unseen data and effectively classify network traffic instances.

The detection rate during training was 99.79%, indicating that the model successfully identified 99.79% of the actual attacks present in the training dataset. Similarly, during testing, the model achieved a detection rate of 99.67%, indicating its capability to accurately identify attacks in the testing dataset. The precision of the SWM-IDS model during training was 99.85%, suggesting that among the instances classified as attacks by the model, 99.85% were indeed true positives. During testing, the precision remained high at 99.70%, indicating a low rate of false positives in the predictions made by the model. The F1-score was 99.80% during training. It provides a balance between precision and recall. The F1-score during testing was 99.65%, further confirming the model's ability to achieve high value in both recall and precision on unseen data. The FPR during training was 0.32%, indicating that only 0.32% of the instances classified as non-attacks were false positives. However, during testing, the FPR slightly increased to 0.41%, indicating a slightly higher rate of false positives compared to the training phase but still maintaining a relatively low false positive rate overall. Figure 8 depicts the graphical plot of results using CIC-IDS-2018 dataset.

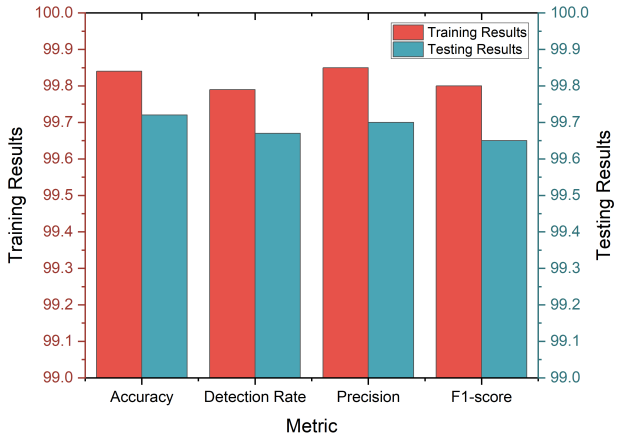


**FIGURE 8.** Graphical plot of SWM-IDS model's results on CIC-IDS dataset.

Table 4 presents the results of the SWM-IDS model on the CIC-DDoS-2019 data set. The SWM-IDS model achieved an accuracy of 99.76% during training, indicating that it correctly classified 99.76% of the samples in the training

**TABLE 4.** Results of the SWM-IDS model on CIC-DDoS dataset.

| Metric | Training Results | Testing Results |
|---|---|---|
| Accuracy | 99.76 | 99.64 |
| Detection Rate | 99.68 | 99.51 |
| Precision | 99.80 | 99.67 |
| F1-score | 99.65 | 99.58 |
| FPR | 0.48 | 0.60 |

dataset. During testing, the accuracy slightly decreased to 99.64%, indicating that the model performed slightly worse on unseen data compared to the training data. A high detection rate indicates that the model is effectively capturing instances of the target class. Here, the training detection rate is 99.68%, and the testing detection rate is 99.51%, both indicating a high level of effectiveness in identifying instances of attacks. The training precision is 99.80%, and the testing precision is 99.67%, demonstrating a high level of precision in both training and testing phases. The training F1-score is 99.65%, and the testing F1-score is 99.58%, both showing high performance in terms of both precision and recall. A lower FPR indicates a better ability to avoid false alarms. In this case, the FPR during training is 0.48%, and during testing, it slightly increases to 0.60%, indicating a slight increase in false positives on unseen data compared to the training data. Figure 9 depicts the graphical plot of results using CIC-DDoS-2019 dataset. Figure 10 represents the plot of the FPR results of the research model on both datasets.

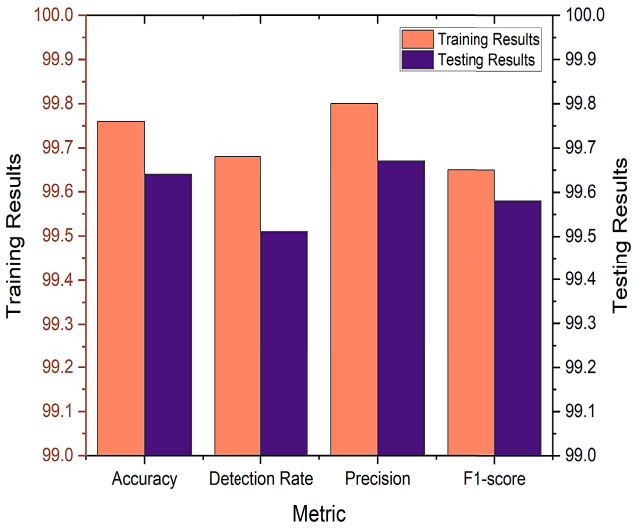


**FIGURE 9.** Graphical plot of SWM-IDS model's results on CIC-DDoS dataset.

Table 5 presents the comparison of the results of the proposed SWM-IDS model with current models. The SWM-IDS model attained an accuracy of 99.72%, demonstrating that it properly classified the instances in the dataset. The percentage difference between the SWM-IDS model and other models varies. For instance, the Split Learning model has the highest accuracy of 99.23%, which is 0.49% lower than the SWM-IDS model's accuracy. Conversely, the Federated Learning model has an accuracy of 98.99%, which

**TABLE 5. Performances comparison of the SWM-IDS model with existing models.**

| Models | Accuracy (%) | DR (%) | Precision (%) | F1-score (%) |
|---|---|---|---|---|
| DNN [14] (Simulated Data) | 93.74 | 93.82 | 93.71 | 93.47 |
| Decision Tree (Bot-IoT) [15] | 97.90 | 93.00 | 93.00 | 93.00 |
| Federated Learning (NSL-KDD) [17] | 98.99 | 93.22 | 99.32 | 98.24 |
| Split Learning (NSL-KDD) [17] | 99.23 | 98.68 | 99.64 | 99.29 |
| BODL-ADC (UCI SECOM) [18] | 99.14 | 94.70 | 95.69 | 95.16 |
| DNN (DS2OS) [19] | 98.28 | 98.00 | 97.00 | 98.00 |
| DDAD-SOEL (Bot-IoT) [21] | 99.68 | 99.32 | 99.33 | 99.32 |
| IDPS (Simulated Data) [22] | 91.00 | 92.00 | 99.00 | 97.00 |
| IDCPRO-DLM (CICIDS2017) [23] | 98.53 | 98.82 | 98.64 | 98.59 |
| CNN-LSTM (CICIoT2023) [25] | 98.75 | 98.75 | 98.75 | 98.75 |
| RNN-XGBoost (CICIDS2017) [26] | 99.00 | 99.00 | 96.00 | 96.00 |
| SALMA (NSL-KDD) [27] | 95.04 | 79.59 | 92.05 | 85.37 |
| MLP (CICIDS2018) [37] | 98.50 | 96.30 | 97.40 | 98.50 |
| MLP (CICDDoS2019) [37] | 97.20 | 97.00 | 96.30 | 95.30 |
| Gradient Boost (CICIDS2018) [37] | 98.70 | 98.40 | 98.60 | 98.10 |
| Gradient Boost (CICDDoS2019) [37] | 99.20 | 98.20 | 98.70 | 98.00 |
| XGBoost (CICIDS2018) [37] | 99.70 | 98.90 | 99.20 | 98.50 |
| XGBoost (CICDDoS2019) [37] | 99.50 | 98.70 | 99.60 | 98.20 |
| SWM-IDS (CIC-IDS-2018) | 99.72 | 99.69 | 99.70 | 99.65 |
| SWM-IDS (CIC-DDoS-2019) | 99.64 | 99.51 | 99.67 | 99.58 |

is 0.73% lower than the SWM-IDS model. The SWM-IDS model achieved a detection rate of 99.69%, indicating that it successfully identified the actual attacks. The percentage differences in detection rate between the SWM-IDS model and other models vary from 0.69% to 20.10%. The SWM-IDS model achieved a precision of 99.70%, indicating that the instances classified as attacks by the model were true positives. The percentage differences in precision between the SWM-IDS model and other models vary from 0.37% to 7.65%. The SWM-IDS model achieved an F1-score of 99.65%, indicating a balance between precision and recall. The percentage differences in F1-score between the SWM-IDS model and other models vary from 0.33% to 14.28%. Overall, the results demonstrate that the SWM-IDS model achieved exceptional performance in terms of accuracy, detection rate, precision, and F1-score, both during training and testing phases, indicating its effectiveness in accurately detecting and classifying network attacks while minimizing false alarms.

Compared with the current models which utilized the CIC-IDS 2018 and CICDDoS 2019 datasets, the research model has outperformed in comparison with both the datasets results. By comparing the research model's results on CIC-IDS-2018 with other models like MLP, gradient boost, and XGBoost, the accuracy of the research model has an improved performance with 0.02% to 1.22%. The SWM-IDS model has a detection rate difference of 0.79% to 3.39%, 0.5% to 2.3% difference in precision, and 1.15% to 1.55% difference in f1-score. By comparing the research model's results on CIC-DDoS-2019 with other models like MLP, gradient boost, and XGBoost, the accuracy of the research model has an improved performance with 0.14% to 2.44%. The SWM-IDS model has a detection rate difference of 0.81% to 2.51%, 0.07% to 3.37% difference in precision, and 1.38% to 4.28% difference in f1-score. Figure 11 depicts the visual representation of the results comparison.

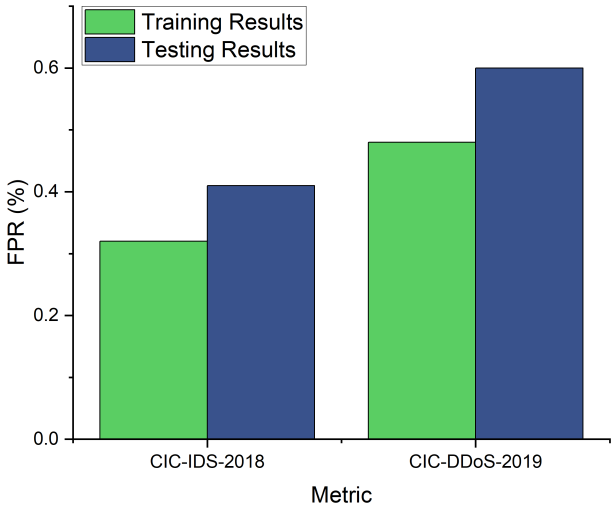


**FIGURE 10. Graphical plot of SWM-IDS model's FPR results.**

The IoT utilizing the GraphSAGE graph neural network model has various benefits. First, GraphSAGE helps the model capture the complex linkages and dependencies between IoT devices and the smart waste management ecosystem. Intrusion detection is more extensive and accurate using this method, improving SWM infrastructure security. Since the CIC-DDoS-2019 dataset is designed to examine DDoS attacks, a common hazard in IoT contexts, the model is robust and generalizable. By selecting the most relevant information for categorization, feature selection methods like the BWOA improve intrusion detection efficiency and efficacy. The research provided a smart and adaptive solution to SWM platforms' IoT intrusion detection security problems.

However, despite its advantages, the proposed research encounter certain limitations. One potential limitation lies in the scalability and computational complexity of the GraphSAGE model, especially when applied to large-scale SWM infrastructures with numerous interconnected IoT devices.

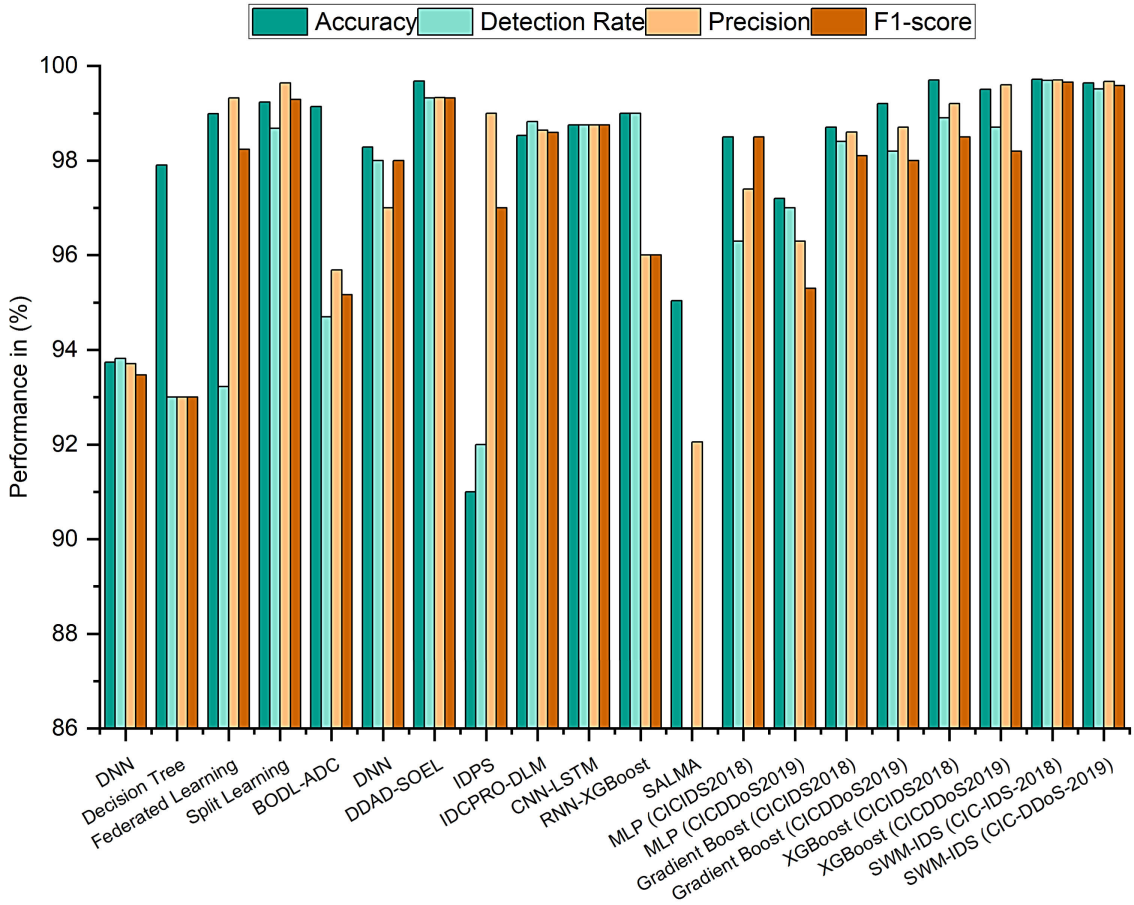


**FIGURE 11.** Graphical plot of results comparison.

Moreover, while the use of the CIC-DDoS-2019 dataset offers valuable insights into DDoS attack patterns, it may not fully capture the diversity and complexity of intrusion scenarios encountered in real-world SWM deployments. As a result, the model's performance, and generalizability in detecting novel and evolving threats outside the scope of the dataset could be limited.

## VII. CONCLUSION

This research proposed a novel IDS model for detecting attacks in a SWM, an IoT-based framework developed using GNN. The research model SWM-IDS includes data collection, preprocessing, selection of features and classification tasks for the identification of attacks. This research model was a binary classification model, which identified the intrusions as either attack or normal. For data collection, the model utilized the CIC-DDoS-2019 and CIC-IDS-2018 data sets, which were employed for training and evaluating the research model. The datasets were split into 70:30 ratio. Initially, the datasets were preprocessed and normalized using data cleaning, unnecessary data removal and Min-Max normalization. After preprocessing, the features were selected using the BWO technique, in which the significant optimal features were selected using the binary vector representation. The GraphSAGE model utilized the chosen characteristics to accurately classify the input instances and identify any potential attacks. The performance of the SWM-IDS model was computed in terms of detection rate, accuracy, FPR, f1-score, and precision. The model attained 99.72% accuracy, 99.69% detection rate, 99.65% f1-score, and 99.70% precision for the CIC-IDS-2018 dataset and attained 99.64% accuracy, 99.51% detection rate, 99.67% precision, and 99.58% f1-score for the CIC-DDoS dataset. These results were compared and validated with other models discussed in the literature review, and as compared, the research model outperformed all the other models. Overall, the results demonstrated that the SWM-IDS model achieved exceptional performanced in terms of accuracy, detection rate, precision, and F1-score, both during training and testing phases, indicating its effectiveness in accurately detecting and classifying network attacks while minimizing false alarms.

Future works for this research will focus on exploring the integration of real-time anomaly detection mechanisms and adaptive learning algorithms to enhance the SWM-IDS model's responsiveness to emerging IoT threats. Investigating the applicability of federated learning techniques to enable collaborative intrusion detection across distributed SWM networks while preserving data security.

## REFERENCES

[1] A. Hassebo and M. Tealab, "Global models of smart cities and potential IoT applications: A review," *IoT*, vol. 4, no. 3, pp. 366–411, Aug. 2023.

[2] R. P. Janani, K. Renuka, and A. Aruna, "IoT in smart cities: A contemporary survey," *Global Transitions Proc.*, vol. 2, no. 2, pp. 187–193, Nov. 2021.

[3] F. A. Turjman, H. Zahmatkesh, and R. Shahroze, "An overview of security and privacy in smart cities' IoT communication," *Trans. Emerg. Telecommun. Technol.*, vol. 33, no. 3, 2022, Art. no. e3677.

[4] D. Eckhoff and I. Wagner, "Privacy in the smart city—Applications, technologies, challenges, and solutions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 489–516, 1st Quart., 2018.

[5] T. Ali, M. Irfan, A. S. Alwadie, and A. Glowacz, "IoT-based smart waste bin monitoring and municipal solid waste management system for smart cities," *Arabian J. Sci. Eng.*, vol. 45, pp. 10185–10198, Jun. 2020.

[6] J. L. Hernandez-Ramos, J. A. Martinez, V. Savarino, M. Angelini, V. Napolitano, A. F. Skarmeta, and G. Baldini, "Security and privacy in Internet of Things-enabled smart cities: Challenges and future directions," *IEEE Secur. Privacy*, vol. 19, no. 1, pp. 12–23, Jan. 2021.

[7] A. K. M. B. Haque, B. Bhushan, and G. Dhiman, "Conceptualizing smart city applications: Requirements, architecture, security issues, and emerging trends," *Expert Syst.*, vol. 39, no. 5, Jun. 2022, Art. no. e12753.

[8] M. F. Elrawy, A. I. Awad, and H. F. A. Hamed, "Intrusion detection systems for IoT-based smart environments: A survey," *J. Cloud Comput.*, vol. 7, no. 1, pp. 1–20, 2018.

[9] P. M. Rao and B. D. Deebak, "Security and privacy issues in smart cities/industries: Technologies, applications, and challenges," *J. Ambient Intell. Hum. Comput.*, vol. 14, no. 8, pp. 10517–10553, 2023.

[10] M. Whaiduzzaman, A. Barros, M. Chanda, S. Barman, T. Sultana, M. S. Rahman, S. Roy, and C. Fidge, "A review of emerging technologies for IoT-based smart cities," *Sensors*, vol. 22, no. 23, p. 9271, Nov. 2022.

[11] K. Zhang, J. Ni, K. Yang, X. Liang, J. Ren, and X. S. Shen, "Security and privacy in smart city applications: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 122–129, Jan. 2017.

[12] D. Singh, B. Pati, C. R. Panigrahi, and S. Swagatika, "Security issues in IoT and their countermeasures in smart city applications," in *Advanced Computing and Intelligent Engineering*. Singapore: Springer, 2018, pp. 301–313.

[13] A. A. Elsaeidy, A. Jamalipour, and K. S. Munasinghe, "A hybrid deep learning approach for replay and DDoS attack detection in a smart city," *IEEE Access*, vol. 9, pp. 154864–154875, 2021.

[14] A. Awajan, "A novel deep learning-based intrusion detection system for IoT networks," *Computers*, vol. 12, no. 2, p. 34, Feb. 2023.

[15] M. M. Alshahrani, "A secure and intelligent software-defined networking framework for future smart cities to prevent DDoS attack," *Appl. Sci.*, vol. 13, no. 17, p. 9822, Aug. 2023.

[16] E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, "Anomaly detection IDS for detecting DoS attacks in IoT networks based on machine learning algorithms," *Sensors*, vol. 24, no. 2, p. 713, Jan. 2024.

[17] I. Priyadarshini, "Anomaly detection of IoT cyberattacks in smart cities using federated learning and split learning," *Big Data Cognit. Comput.*, vol. 8, no. 3, p. 21, Feb. 2024.

[18] P. Manickam, M. Girija, S. Sathish, K. V. Dudekula, A. K. Dutta, Y. A. M. Eltahir, N. M. A. Zakari, and R. Gilkaramenthi, "Billiard based optimization with deep learning driven anomaly detection in Internet of Things assisted sustainable smart cities," *Alexandria Eng. J.*, vol. 83, pp. 102–112, Nov. 2023.

[19] D. K. Reddy, H. S. Behera, J. Nayak, P. Vijayakumar, B. Naik, and P. K. Singh, "Deep neural network based anomaly detection in Internet of Things network traffic tracking for the applications of future smart cities," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 7, Jul. 2021, Art. no. e4121.

[20] R. Bukhowah, A. Aljughaiman, and M. M. H. Rahman, "Detection of DoS attacks for IoT in information-centric networks using machine learning: Opportunities, challenges, and future research directions," *Electronics*, vol. 13, no. 6, p. 1031, Mar. 2024.

[21] M. Aljebreen, H. A. Mengash, M. A. Arasi, S. S. Aljameel, A. S. Salama, and M. A. Hamza, "Enhancing DDoS attack detection using snake optimizer with ensemble learning on Internet of Things environment," *IEEE Access*, vol. 11, pp. 104745–104753, 2023.

[22] A. Bhardwaj, S. Bharany, A. W. Abulfaraj, A. O. Ibrahim, and W. Nagmeldin, "Fortifying home IoT security: A framework for comprehensive examination of vulnerabilities and intrusion detection strategies for smart cities," *Egyptian Informat. J.*, vol. 25, Mar. 2024, Art. no. 100443.

[23] F. S. Alrayes, M. M. Asiri, M. Maashi, A. S. Salama, M. A. Hamza, S. S. Ibrahim, A. S. Zamani, and M. I. Alsaid, "Intrusion detection using chaotic poor and rich optimization with deep learning model for smart city environment," *Sustainability*, vol. 15, no. 8, p. 6902, Apr. 2023.

[24] D. Li, L. Deng, M. Lee, and H. Wang, "IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning," *Int. J. Inf. Manag.*, vol. 49, pp. 533–545, Dec. 2019.

[25] S. Yaras and M. Dener, "IoT-based intrusion detection system using new hybrid deep learning algorithm," *Electronics*, vol. 13, no. 6, p. 1053, 2024.

[26] S. Yadav, H. Hashmi, and D. Vekariya, "Mitigation of attacks via improved network security in IoT network environment using RNN," *Meas., Sensors*, vol. 32, Apr. 2024, Art. no. 101046.

[27] H. Ali, O. M. Elzeki, and S. Elmougy, "Smart attacks learning machine advisor system for protecting smart cities from smart threats," *Appl. Sci.*, vol. 12, no. 13, p. 6473, Jun. 2022.

[28] R. Ma, X. Chen, and R. Zhai, "A DDoS attack detection method based on natural selection of features and models," *Electronics*, vol. 12, no. 4, p. 1059, 2023.

[29] S. Aktar and A. Y. Nur, "Towards DDoS attack detection using deep learning approach," *Comput. Secur.*, vol. 129, Jun. 2023, Art. no. 103251.

[30] M. Alanazi and A. Aljuhani, "Anomaly detections for Internet of Things cyberattack," *Comput., Mater. Continua*, vol. 72, no. 1, pp. 261–279, 2022.

[31] S. Bacha, A. Aljuhani, K. B. Abdellafou, O. Taouali, N. Liouane, and M. Alazab, "Anomaly-based intrusion detection system in IoT using kernel extreme learning machine," *J. Ambient Intell. Humanized Comput.*, vol. 15, no. 1, pp. 231–242, 2024.

[32] F. Khan, A. A. Al-Atawi, A. Alomari, A. Alsirhani, M. M. Alshahrani, J. Khan, and Y. Lee, "Development of a model for spoofing attacks in Internet of Things," *Mathematics*, vol. 10, no. 19, p. 3686, 3686.

[33] M. A. Tawhid and A. M. Ibrahim, "Feature selection based on rough set approach, wrapper approach, and binary whale optimization algorithm," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 3, pp. 573–602, Mar. 2020.

[34] J. Liu, G. P. Ong, and X. Chen, "GraphSAGE-based traffic speed forecasting for segment network with sparse data," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 1755–1766, Mar. 2022.

[35] H. F. Eid, "Binary whale optimisation: An effective swarm algorithm for feature selection," *Int. J. Metaheuristics*, vol. 7, no. 1, pp. 67–79, 2018.

[36] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representations learning on large graph," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[37] N. Aslam, S. Srivastava, and M. M. Gore, "ONOS DDoS defender: A comparative analysis of existing DDoS attack datasets using ensemble approach," *Wireless Pers. Commun.*, vol. 133, no. 3, pp. 1805–1827, Dec. 2023.

**MAJED M. ABOROKBAH** (Member, IEEE) received the B.Sc. degree from Taif University, Saudi Arabia, the M.Sc. degree from Bradford University, U.K., and the Ph.D. degree from De Montfort University, U.K. He is currently the Dean of the Faculty of Computers and Information Technology, University of Tabuk, Saudi Arabia. His research interests include artificial intelligence, software engineering, context-aware systems, cybersecurity, and steganography. He has contributed significantly to these fields and has published numerous papers in international journals and conferences. Additionally, he has played an active role in organizing various workshops and conferences related to his research areas. He has also made notable contributions to the establishment of the Robotics Center, University of Tabuk.