Received: 13 December 2023 | Revised: 20 September 2024 | Accepted: 26 October 2024

IET Wireless Sensor Systems

DOI: 10.1049/wss2.12100

The Institution of Engineering and Technology WILEY

REVIEW

Intrusion detection in cluster-based wireless sensor networks: Current issues, opportunities and future research directions

Ayuba John[1,2] | Ismail Fauzi Bin Isnin[2] | Syed Hamid Hussain Madni[3] | Muhammed Faheem[4,5]

[1]Faculty of Computing, Federal University Dutse, Dutse, Nigeria

[2]Faculty of Computing, Universiti Teknologi Malaysia (UTM), Johor Bahru, Malaysia

[3]School of Electronic & Computer Science, University of Southampton, Johor Bahru, Malaysia

[4]School of Technology and Innovations, University of Vaasa, Vaasa, Finland

[5]VTT Technical Research Centre of Finland Ltd., Espoo, Finland

Correspondence

Muhammed Faheem.
Email: muhammad.faheem@wasa.fi

Funding information

Universiti Teknologi Malaysia; Vaasan yliopisto

Abstract

Wireless sensor network (WSN) cluster-based architecture is a system designed to control and monitor specific events or phenomena remotely, and one of the important concerns that need quick attention is security risks such as an intrusion in WSN traffic. At the same time, a high-level security method may refer to an intrusion detection system|intrusion detection systems (IDS), which may be employed effectively to achieve a higher level of security in detecting an intruder attack or any attack initiated within a WSN system. The significance of the detection of network intrusions on heterogeneous cluster-based sensor networks with wireless connections, as well as the approaches to machine learning utilised in IDS model development, were discussed. In addition, this research conducted several comparative studies of feature selection techniques and machine learning methodologies in the development of intrusion detection systems. The authors used a bibliometric indicator to identify the leading trends when it comes to IDS, and the VOS viewer was used to create a spatial mapping of co-authorship, co-occurrence, and citation types of analysis with their respective units of study. The purpose of this research paper is to generate relevant findings and a research problem formulation that can lead to a research gap in the research topic's domain area.

KEYWORDS

Cybersecurity, internet of things, intrusion detection, machine learning, wireless sensor networks

1 | INTRODUCTION

The clustering architecture has numerous advantages in WSNs. It is capable, for example, of reducing the size of inter-node communication by focussing on data transmission within designed clusters and decreasing the amount of transmissions to the base station [1]. However, security attacks have been a major concern in both homogeneous and heterogeneous WSNs, according to John and Igimoh [2]. Attackers may deliberately explore the target system's vulnerabilities and execute different kinds of threats in order to gain access to the system, some of which may reveal sensitive information [3]. Unfortunately, as attackers become more sophisticated, new threats and vulnerabilities emerge at a rapid pace [4]. An intrusion detection system IDS is a tried-and-true mechanism for dealing with malicious threats in WSNs [5]. An IDS is a software or hardware system that manages and recognises intrusions on a regular basis and alerts the computer or system administrator to take a predetermined action. This alert report supports the system administrator or operator in detecting the susceptibility in the system and resolving it [6]. According to Li and Qin [7], Anderson proposed the knowledge of IDS in the 1980s, and it can protect the network or host from attacks, their design and construction differ.

Because WSN is susceptible to a diversity of attacks, most of the IDS systems that were proposed currently show some strength. Still, unfortunately, the majority of the systems generate computational overhead and consume sensor node resources [8]. Sensor node to sensor node, sensor node as a member of the cluster head, cluster to cluster, and finally,

cluster head to base station comprise the connection in Cluster-Based Wireless Sensor Networks (CBWSN) [9]. Figure 1 depicts a single-hop communication, such as one from a sensor node member to the cluster head, or as shown in Figure 2, multi-hop communication can occur, particularly from a sensor node member to the cluster head and from the cluster head to the base station. Thus, inter-cluster communication refers to communication within the same cluster [10], whereas intra-cluster communication relates to communication between clusters [11]. This paper conducted a comparative analysis of detection systems for network intrusions to show the advantages and disadvantages of methods for selecting features used with classification algorithms, as well as the prospects for future research directions. A bibliometric analysis was carried out to properly investigate the research area. We performed a comparative analysis investigation into network intrusion detection systems based on their feature selection approaches and classifier algorithms.

The following are the primary contributions of this paper:

i. We presented a taxonomy of CBWSN that depicted the major research challenges, clustering methodologies, network design, and packet delivery mode.
ii. We compared several feature selection strategies used in the creation of intrusion detection systems and evaluated the efficacy of machine learning algorithm approaches in IDS models.
iii. In the development of intrusion detection systems, we conducted a comparative assessment of the utility of autoencoders for dimensionality reduction.
iv. To aid in the identification of researchable keywords in the domain area, we performed a bibliometric analysis of intrusion detection systems in WSNs.
v. We generated relevant findings and formulated the research challenge, which contributed to identifying the research gap in the domain area of the research topic.

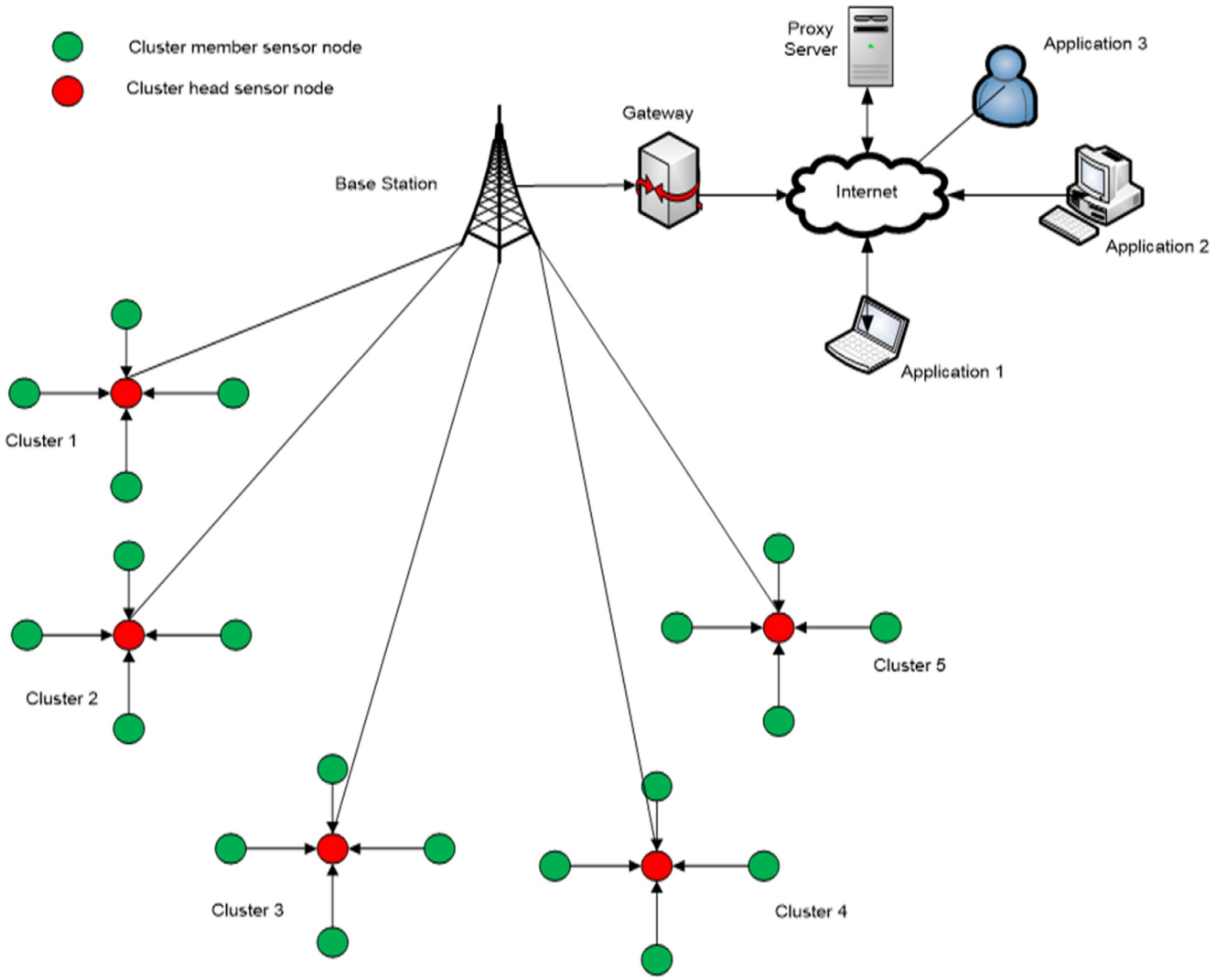

FIGURE 1 Wireless sensor networks (WSNs) with a single hop in clusters.

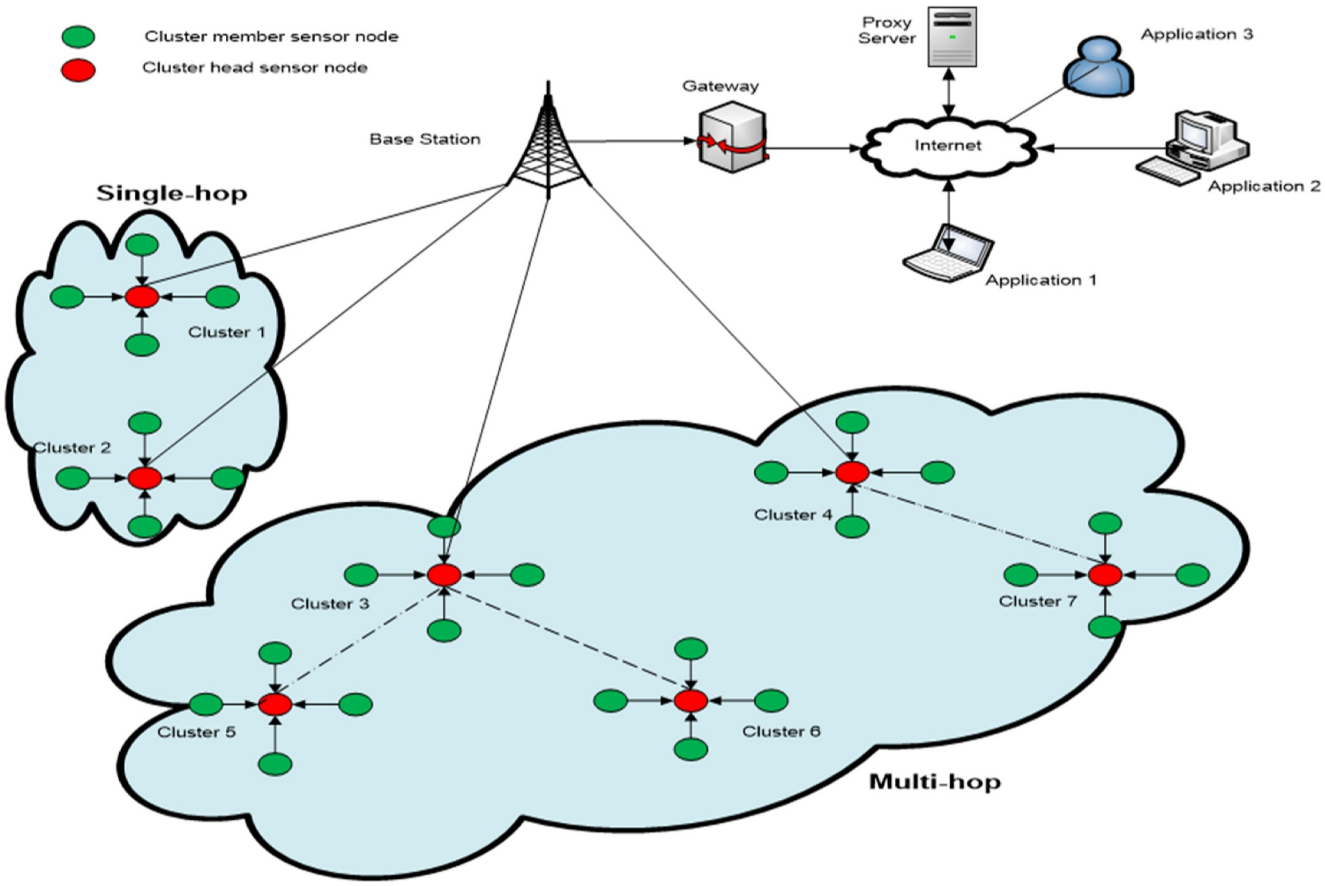

FIGURE 2 Wireless sensor networks (WSNs) with multiple hops in a cluster.

The remaining sections of the paper are as follows: Section 2 discusses the related studies in CBWSN, Section 3 discusses clustering WSNs, Section 4 discusses intrusion detection systems as security mechanisms used in clustering WSNs, Section 5 discusses on the IDS on heterogenous CBWSN, Section 6 discusses on the approaches of machine learning for IDS. Section 7 describes the bibliometric analysis of the reviewed related studies, Section 8 discusses on the challenges and the future research directions, Section 9 discusses on the research findings and the remainder of the section presents the conclusion, acknowledgement and references.

2 | RELATED WORKS ON CLUSTER-BASED WIRELESS SENSOR NETWORKS

Clustering architecture in WSN has aided in improving service quality by resolving some of the network's common challenges with fewer human efforts. Prabha, Darly [12], to improve sensor node energy efficiency without sacrificing data transmission, propose a tridiagonal block matrix with heterogeneous clustering in an improved hierarchical compressive sensing WSN. In order to maximise cost efficiency, the block matrix is used to select nodes based on signal strength. Kowsalya and Jeetha [13] provided a simple model cryptographic data aggregation algorithm with a secure lightweight to provide an improved computational effectiveness and confidentiality shield with access control in a cluster WSN. Sert and Yazici [14] claim that trial-and-error methods are used by field experts in almost all cases of fuzzy rule-based clustering systems, making an optimal fuzzy system nearly impossible to achieve, and propose an improved clonal collection algorithm to boost the energy efficiency of fuzzy rule-based clustering algorithms. Sharma and Vashisht [15] presented a modified invasive weed optimisation-based clustering algorithm that emphasises node fittest selection using a fuzzy inference model for energy efficiency and network life. Fanian and Rafsanjani [16] propose a protocol to improve network lifetime by utilising fuzzy multi-hop clustering and the number of node packets received based on application features.

The authors in refs.[17, 18] proposed a collective model based on machine learning and meta-heuristics to address the high computational time and extra overhead delay in the meta-heuristics model. It succeeded in extending the network lifetime and is determined by application-specific requirements, but it necessitates more offline computation time. In order for the lifetime of the WSNs to be improved and increase network coverage, Bohra and Kumar [19] enhanced fuzzy rules and ranges of linguistic variables to avoid isolated nodes during cluster head election, and [20–22] presented a novel clustering method that decreases the rate of energy consumption and ensures network stability for a

long time. Gbadouissa and Ari [23] enhance the energy efficiency of WSN nodes. Researchers used hypergraph theory in conjunction with heuristic clustering, and Ramani and Amarendra [24] improved the network lifetime by using machine learning to predict the shortest path. [25, 26] To prevent malicious nodes from becoming cluster heads, a trust estimation device was proposed at two levels: the cluster member level as well as a local clustering algorithm, which increased the detection rate of suspicious nodes. Deepa and Suguna [27] used a path selection algorithm known as a round-robin to transfer data packets to the sink and to reduce transmission delay and communication overhead. Prakash and Kavitha [28] developed a Particle swarm optimisation and enhanced ant colony optimisation are combined in a hybrid heuristic data aggregation protocol for clustering the WSN to maximise network lifetime by harmonising energy and reducing delay because the ant colony optimisation algorithm can perform more precise searches, but at a slower rate. Meanwhile, the particle swarm optimisation algorithm can search more quickly but with less precision. In summary, some of the common challenges identified by the researchers above and solutions provided include the cost of energy consumption by nodes, node latency, cluster head selection of nodes, bottleneck and poor scalability of nodes; and packet collision.

3 | CLUSTERING WIRELESS SENSOR NETWORK

3.1 | Taxonomy

Figure 3 below presents a taxonomy of CBWSN that depicts the major research challenges, clustering methodologies, network design, and packet delivery mode, described as follows:

i. *Key research issues:* In WSNs, clustering has a broader range of applications, including data aggregation, energy efficiency, load balancing, network lifetime, fault tolerance, network scalability, reliability, network security, packet transmission delay, and routing algorithm is required for communicating nodes to transmit packets [29] successfully
ii. *Optimised clustering algorithm techniques:* As shown in Figure 3, a variety of optimised clustering algorithm approaches, particularly k-means, genetic algorithms, neural networks, reinforcement learning, swarm intelligence, and fuzzy c-means, have been used to build CBWSN [29]. Fuzzy logic approaches have been clearly used for exploring and determining composite multiple activities owing to their ability to convey approximate human reasoning and suitability in WSN applications that do not require a huge sum of nodes within the goal arena [30]. According to Amutha, Sharma [30], K-means clustering algorithm consists of four different types of clustering methods: Centroid-based clustering, which arranges data into non-hierarchical clusters, is effective but subtle in early situations and outliers; Density-based clustering, which connects areas of high density into clusters but does not assign outliers to clusters; it has difficulty with data of varying densities and high dimensions; Distribution-based clustering, which assumes that data is made up of distributions and that as one's distance from the distribution's centre increases, the probability that a point belongs to the distribution decreases.
For hierarchical data, hierarchical clustering, which generates a tree of clusters, is well suited [29] on overlapping datasets; the fuzzy C-means clustering algorithm performs better, with each data point belonging to multiple clusters and accompanied by its probability score or likelihood [29]. Unlike the k-means clustering algorithm, the genetic clustering algorithm uses a sample of numerous solutions to find solutions to a given problem, making some random changes to the solution in multiple directions, thus reaching an optimal solution without stopping at the first available solution [30]. The neural network clustering algorithm evaluates the centroids of associated cluster collections in training data and is a fast and stable clustering algorithm. It converges faster because it does not require a whole sum of clustering iterations. The Swarm Intelligence Clustering Algorithm groups all nodes in a cluster, with each node having its own problem to solve independently of what the other nodes are doing [31].
iii. *Network architecture:* There are three major network architectures used in CBWSN: flat-based, location-based, and hierarchical-based approaches, which are mostly considered in large-scale WSNs due to their improved network lifetime performance. Because of the variety of WSN applications, choosing an appropriate network architecture for a clustering solution remains difficult [32].
iv. *Packet delivery mode:* Communications in WSNs involve data flows [33], which are referred to as packet delivery modes; the mode can be in real-time for online operations in real-time systems, and the majority of WSNs operate in real-time, with the exception of a few that operate offline and are classified as non-real-time delivery.

3.2 | Characteristics

Cluster-Based Wireless Sensor Networks' sensor nodes are classified based on their type, role, location awareness, cluster count, inter- and intra-cluster communication, cluster head selection, and node mobility [34].

i. *Sensor node type:* When considering sensor nodes based on resource capability, a WSN is divided into two networks: homogeneous and heterogeneous [35]. When all sensor nodes are identical in terms of ability and resources, it is homogeneous; when all of the sensor nodes have different resource capabilities and differ in terms of processing ability, it is heterogeneous.

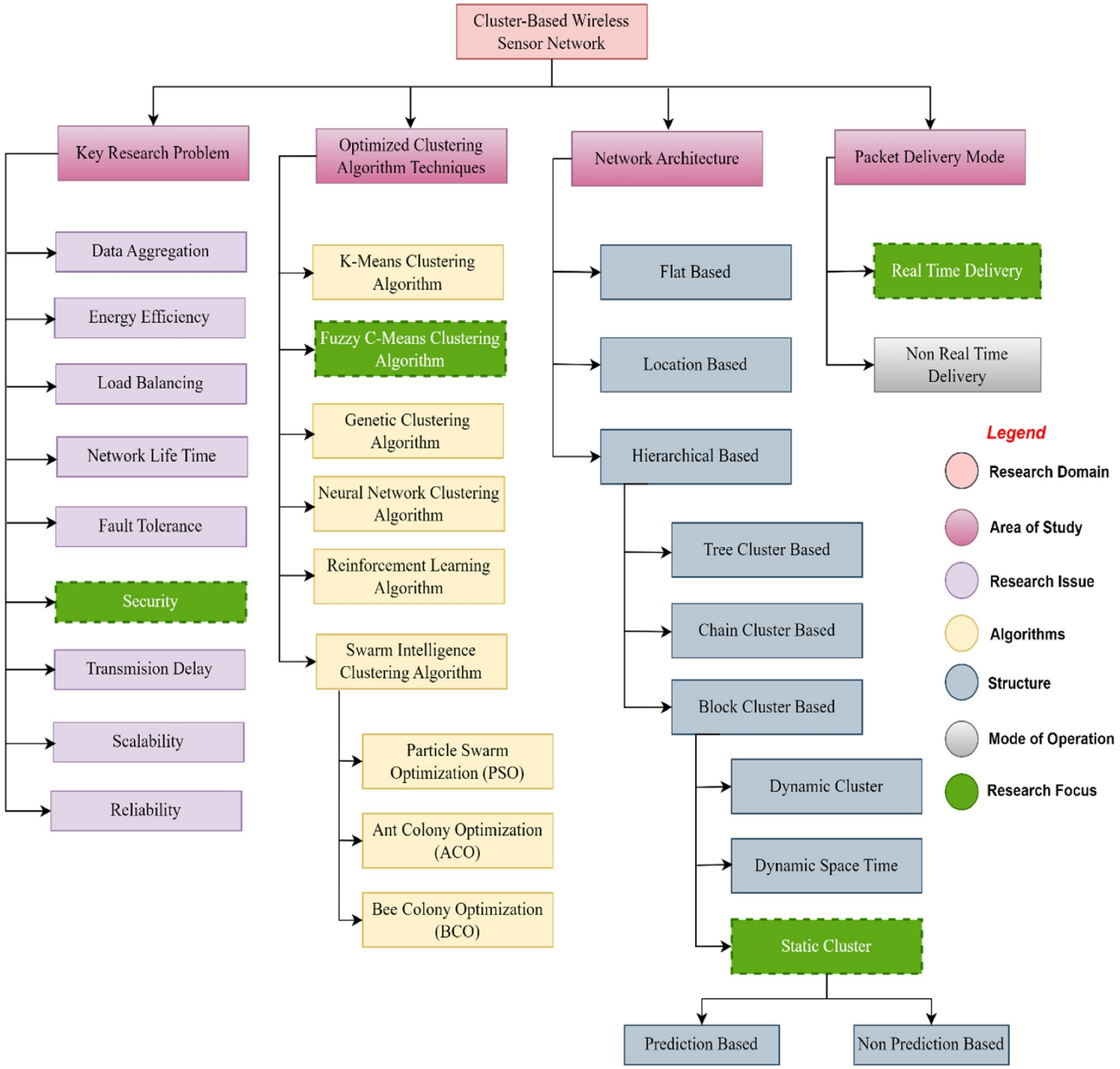

FIGURE 3 The taxonomy of clustering wireless sensor networks (WSNs).

ii. *Role of sensor node:* Nodes in WSNs can also be classified based on their role, which can be either that of a member or that of the cluster head [36].
iii. *Sensor node awareness of location:* Several other clustering algorithms require sensor nodes to know where they are in relation to the base station; as a direct consequence, a global positioning system must be implemented in the network to calculate the distance among both their location and the nodes head and base station as stated by Zakariayi and Babaie [37].
iv. *Cluster count:* The cluster count is a significant characteristic of a WSN; it can determine the clustering technique's effectiveness [38]. Some methods are only effective if a specific number of network clusters are considered, whereas others require a random selection of the cluster count, depending on the application in use.
v. *Communication between clusters and within clusters:* The connection from Sensor node to sensor node, sensor node as a member of the cluster head, cluster to cluster, and finally cluster head to base station are all part of CBWSN. A one-hop communication, such as one from a sensor node to the cluster head, or a multi-hop communication, particularly from a sensor node member to the cluster head, as well as from the cluster head to the base station, can occur [39]. Inter-cluster communication is thus regarded as "cross-communication," whereas intra-cluster communication is considered to be "based on inter-communication."

vi. *Choosing a Cluster Head:* In a cluster network, the head selection can be dynamic in a mobile network or fixed in a static clustering WSN. The distance between nodes, the proximity to the base station, the node's energy efficiency, and the possibility of a cluster head are some of the common parameters considered for head selection [40]. In a sensor network with wireless connections in a mobile cluster, the cluster head rotates on a regular basis based on energy efficiency to make certain that node energy usage is uniform, and the communication protocol may be less efficient as a result of the cluster head being chosen at random.

vii. *The sensors' mobility:* After clustering, the sensor nodes in clustering methods are all stable, and since there is no existing clustering protocol that can facilitate node mobility in static networks of wireless sensors, mobile clustering WSNs can be considered more scalable in terms of the application range to enhance the performance of the network [34].

3.3 | Advantages

i. *Effective resource usage:* Energy consumption is one of the resources that is properly utilised [41]; because data packets from each member node are transmitted to the cluster head at the synchronisation time assigned to it, this helps in reducing collision rates and saves a lifetime's worth of energy [42]. Although communication occurs between the cluster head and each cluster head, there is a small routing table at the base station to facilitate direct communication among cluster members; this has established communication boundaries between clusters [43].

ii. *Efficient architectural management:* Any failure of an individual node is handled at each cluster with no significant impact on the network structure. Because a node failure cannot disrupt the entire network, the cluster head performs periodic re-clustering to address the node failure [29].

iii. *Improve network performance:* Because the communication process is done in two stages: One link connects the member nodes to the cluster head, and two links connect the cluster head to the base station, the network lifetime is greatly improved by reducing the individual node's energy consumption [44]. This process has significantly improved the network's scalability, network coverage area, and rate of packet delivery delays [29, 45].

3.4 | Challenges

Despite the fact that CBWSN have made significant contributions to current WSN application areas such as smart grids, intelligent-based healthcare security monitoring, big data environments, circular economies in smart cities, tracking and tactical surveillance, and green data gathering with the cloud [46], It also has various issues, particularly in data aggregation, network scalability, resource management, and security among others, as detailed below:

i. *Data aggregation:* The deployment of CBWSN in current applications has great advantages, though it has contributed to an increase in the volume of data collection and processing. It requires efficient processing, and it is not easy to utilise the conventional method of data processing on such networks. The high increase in the application of CBWSN involves the deployment of sensors, which are responsible for producing large data volumes, making WSNs key contributors to big data [47, 48]. Big data technology can be used to gather, analyse, store, and transfer data in CBWSN. The main challenge of big data in CBWSN is data aggregation, which involves collecting data from various sources in the network and combining it to eliminate redundancy and reduce the consumption of available network resources. However, the correlation between data aggregation and energy consumption challenges for big sensor data in CBWSN needs to be considered [49]. The cluster head performs the approach of data aggregation in CBWSN; this reduces the bandwidth overhead by reducing the size of the packet transmission per time.

ii. *Network Scalability:* One of the objectives of the clustering WSN is to achieve dynamic scalability of the network in order to improve its performance. The cluster-based WSN scalability determines the stability of the networks, and the cluster should be able to either increase or decrease its cluster members [50]. Many factors can lead to it, especially a sensor node that may fail due to threats on the networks in the harsh environments where it is deployed, which may result in energy depletion of the nodes. At this point, the cluster members will be reduced, which will decrease the network coverage. However, in a situation where nodes are added to the clusters, it will rescale the network and increase the number of cluster heads to expand the network coverage.

iii. *Resource Management:* Resource management in a cluster-based WSN is a big challenge due to the scarcity of sensor nodes, and wireless data transmission is the major energy consumer in the network [51]. CBWSN gather data from each node and transmit it either indirectly or directly to avoid an end-to-end delay. It will require efficient optimisation of the cluster transmission by reducing the number of hops between the nodes and the base station. However, the data from each node is transmitted directly to the base station. It will result in high energy consumption, thus reducing the delay and energy consumption simultaneously [52]. A good routing method will be required to improve the routes and reduce the number of hops. It will require an efficient optimisation of the inter-cluster routing and the intra-cluster routing to determine the shortest routes that can improve efficiency and reduce the delay of the data transmission.

iv. *Security:* Security is one of the key issues in CBWSN due to resource constraints, especially the limited energy, limited computational speed and memory, and limited bandwidth

associated with the network, which determines the uniqueness of the security challenges bedevilling the network. Data transmission is performed in real-time, and in most cases, a multi-hop communication mode is adopted to overcome some of these resource constraints [53]. However, the multi-hop communication mode of the CBWSN is vulnerable to several types of attacks, such as identity tracing and masquerading by the intruder and modification of the data by intermediary nodes (middleman attack), which can be an intruder masquerading to change the routing information between the sensor nodes in the clusters. One of the effective security mechanisms developed by several researchers to handle security challenges in CBWSN is an IDS model that involves feature selection techniques and classification algorithms [54–56].

4 | INTRUSION DETECTION SYSTEMS

A model of an IDS is a system that can evaluate a suspected intruder in network traffic [57] or attacks initiated from within the system after they have occurred and provide a warning sign to the system administrator to take a predefined action programmed into the system as depicted in Figure 4 below. Misuse detection (or signature-based detection), anomaly detection, and specification-based detection techniques are the three types of IDS techniques [58].

4.1 | Misuse detection technique or signature-based detection system

The system uses the signatures to compare the behaviour of an unknown attack pattern, which was defined as a security threat observed and stored in the database [59] in this detection technique. If the behaviour is found to be inconsistent with the patterns stored in the database, the system classifies it as an intrusion into the system; otherwise, it is classified as normal. According to ref. [60], the main disadvantage of this detection technique is that it cannot detect unknown threats or new attack patterns that are not stored in the system's database, so the attack patterns in the database must be consistently and continuously updated. In order to avoid false alarms and malicious tricks, the signatures or patterns written into the database must also include all possible attack patterns of the anticipated security threats.

4.2 | Anomaly detection technique

This detection technique is more concerned with system behaviour than with attack behaviour [61]. The system is trained as an intelligent system in order to establish an automated normal causal behaviour of the system, which will be consistently compared with any activity through the system. If it does not correlate with the system's normal behaviour, it will consider the activity to be an intrusion by triggering an alarm indicated by flag action. According to Lawal and Shaikh [62], this method is capable of detecting any new security threat introduced into the system. However, every unnoticed activity by the system will result in an inaccurate false alarm by indicating anomalous behaviour that may not be required for an intruder to be flagged as an intruder, and in some cases, an intrusion whose behaviour does not identify an Anomalous may not be detected, resulting in false negative alarms [63].

4.3 | Specification-based detection

This type of detection technique focuses on the combined strength or advantages of both the misuse detection and the anomalous detection techniques [64], which are based on deviations from the system's normal behaviour by providing a specification description of the system's operation and observing any contrary behaviour in relation to the specifications. However, in this case, normal behaviour is defined not by machine learning training techniques but by the system specifications, which allow the system to detect both unknown

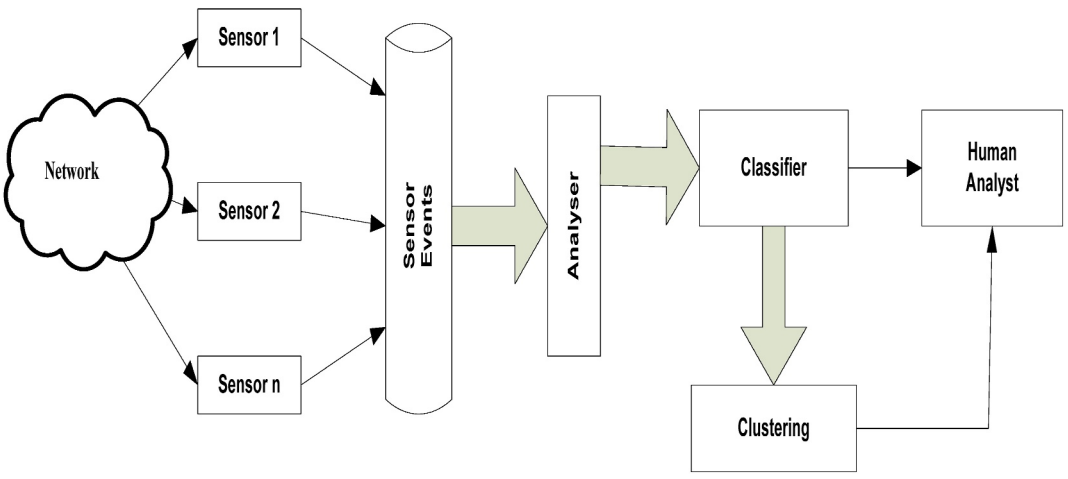

FIGURE 4 Intrusion detection system (IDS) architecture.

attacks and unnoticed behaviour to avoid false alarm triggering. However, because detailed specifications must be implemented manually, there is a significant time and cost disadvantage [65]. Attacks on WSNs can take the form of either an active attack aimed at destroying network assets or a passive attack aimed at stealing valuable information from the networks. A good IDS model can evaluate all suspected attacks as they occur and send a warning signal to alert the system to take predefined actions to protect the network system from intruder or insider attacks [5]. The attacks can be classified into four classes in anomaly detection: Denial of Service (DoS), Probe, User to Root (U2R), and Remote to Local (R2L) attacks [66].

5 | INTRUSION DETECTION SYSTEM SYSTEMS ON HETEROGENOUS CLUSTER-BASED WIRELESS SENSOR NETWORKS

Data collection and routing have been shown to be more efficient when WSNs are clustered. Its benefits include energy efficiency, scalability, even energy distribution, processing power, and increased communication range [29]. Clustering is used in WSNs to address issues such as data aggregation, load balancing, Fault tolerance, network lifetime, energy efficiency, scalability, reliability, QoS support, and security are just a few of the benefits of using a cluster-based WSN [30]. The clustering process is divided into two stages: the initialisation stage and the stability state. The cluster head selection, cluster head role advertisement, cluster formation, and schedule creation are all part of the initialisation stage [42].

When a cluster head is chosen, it will broadcast an advertisement message to the nodes within its communication range, inviting them to join as member nodes, and then create a scheduling broadcast path for all of its members. Afterwards, during the stability state, each cluster member will periodically wake up in its designated slot to transmit its sense packet to the cluster head, which will then forward the aggregated packets from each cluster head to the base station for further processing. Because there are fewer hop connections between nodes and the base station in heterogeneous WSNs, the end-to-end delivery rate is higher than in homogeneous WSNs [29, 45]. Figure 5 below depicts the communication modes in heterogenous CBWSN secure with an IDS. A NIDS is defined

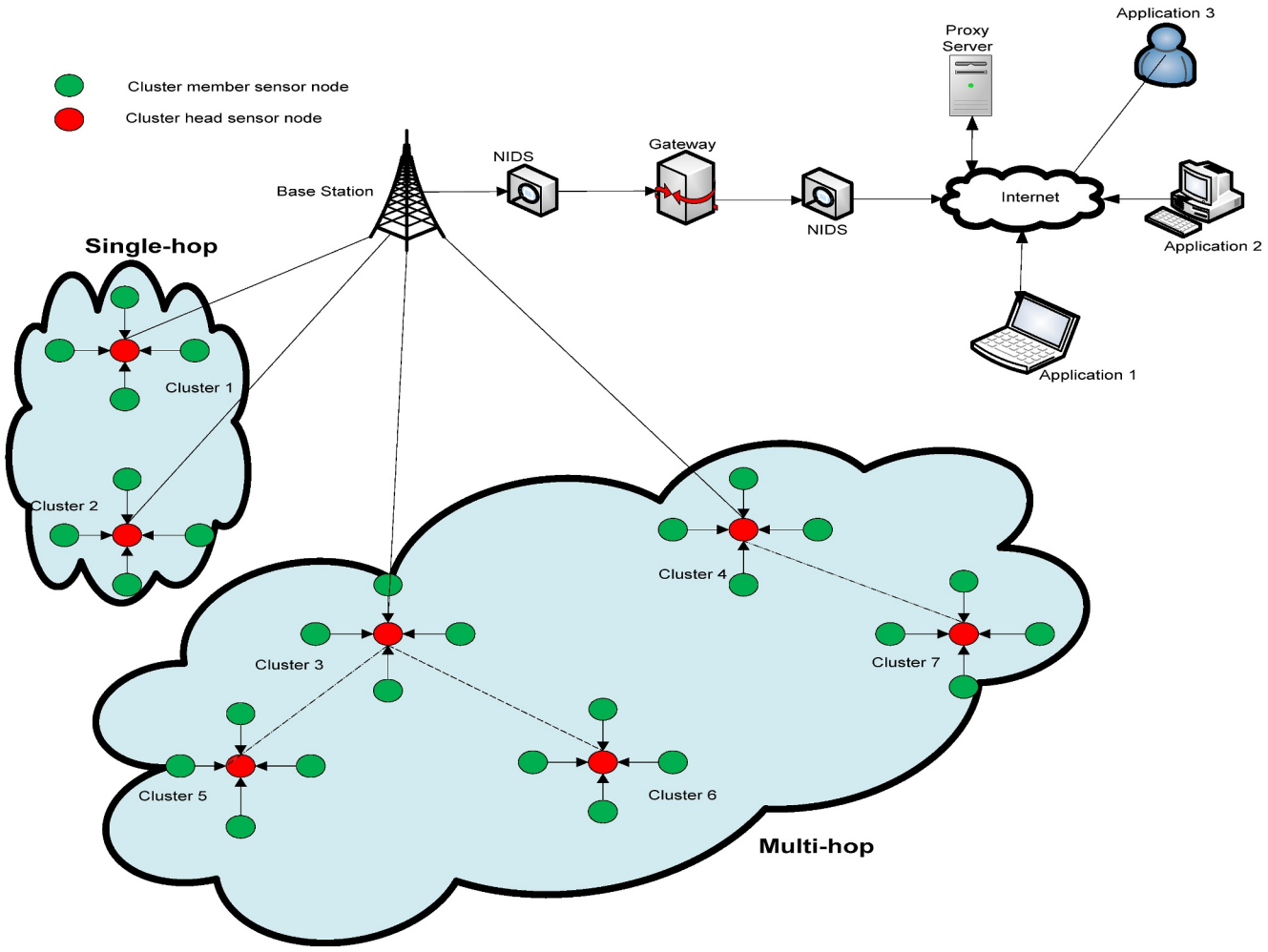

FIGURE 5 An intrusion detection system on a cluster-based wireless sensor network (WSN).

as a hardware or software device that can evaluate any network to detect intrusions and malicious behaviour in order to protect the network from attacks that may originate from within or outside the network so that system managers are notified whenever an abnormality is detected. The system managers will then take a predefined action to secure the network [67]. Network intrusion detection methods are used on devices that can capture network traffic to be monitored [68] as shown in Figure 5.

5.1 | Typical deployment strategy of network intrusion detection system

A simple NIDS deployment strategy is depicted in Figure 6, and the NIDS inspects all network traffic that passes through the switch via its monitoring interface. The NIDS could collect information from the switch via the port or network tap; likewise, it might belong to an internal switch component, particularly the NIDS switch module. The intruder can initiate a succession of malicious actions, such as attempting to send some false network traffic in order to prevent the normal network traffic of the cluster-based WSN from reaching their destination on the network, or the threats can be on the network server by scanning for vulnerability ports through the switch to launch an attack in the network.

In the case of anomaly-based detection, the NIDs can identify the malicious attack by comparing it to normal network traffic behaviour and alerting the network security administrator via the control and reporting interface to take a predefined action, such as isolating the sensor node from the cluster and stopping all traffic coming from it, depending on how the NIDS is configured in the network. It could also instantaneously reset the node connection by incorporating rule sets to the firewall or router that deny the sensor node any access to the cluster-based WSN. In the case of misuse-based detection, after identifying the intrusion, the NIDs will compare the signature of the attack with the attack pattern in the database and take the appropriate network actions. The important aim of the cluster-based WSN depicted in Figure 7 is to manage extracted information before sending it to remote locations. The NIDs are perhaps a possible method for dealing with a broad range of security attacks in the wide network's coverage area.

5.2 | Real-world challenges of intrusion detection systems implementation in cluster-based wireless sensor network

Many researchers have proposed protocols to handle the computational overhead in real-world CBWSN. Still, the majority of these protocols are based on cryptographic techniques due to the limited resources and computation capabilities of the sensor nodes in the CBWSN [69], which may result in high resource consumption, such as energy consumption, for each node by imposing a significant load on the nodes. As a result, some nodes would fail due to energy depletion, affecting the scalability of the cluster-based WSN [29]. It may pose a hurdle to the feasibility of implementing an IDS model in real-world cluster-based WSN applications. Communication in CBWSN is broadcast, making them more vulnerable to malicious attacks, particularly denial-of-service attacks. Cluster heads, who oversee local clusters, are a prime target for adversaries because if they are captured or hacked, a DoS attack can bring the entire cluster down. It clearly demonstrates why CBWSN necessitate the proper installation of an IDS model capable of dealing with attacks on both the host and network infrastructure.

The uniqueness of WSNs, such as limited power supply, low transmission bandwidth, small memory size, and data

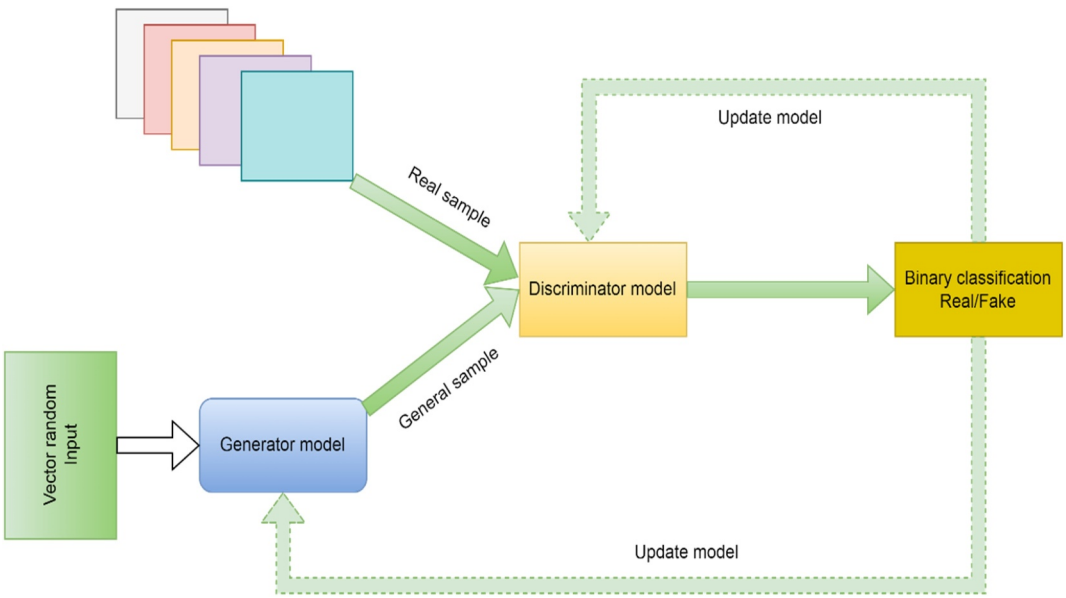

FIGURE 6 Generative adversarial networks (GANs).

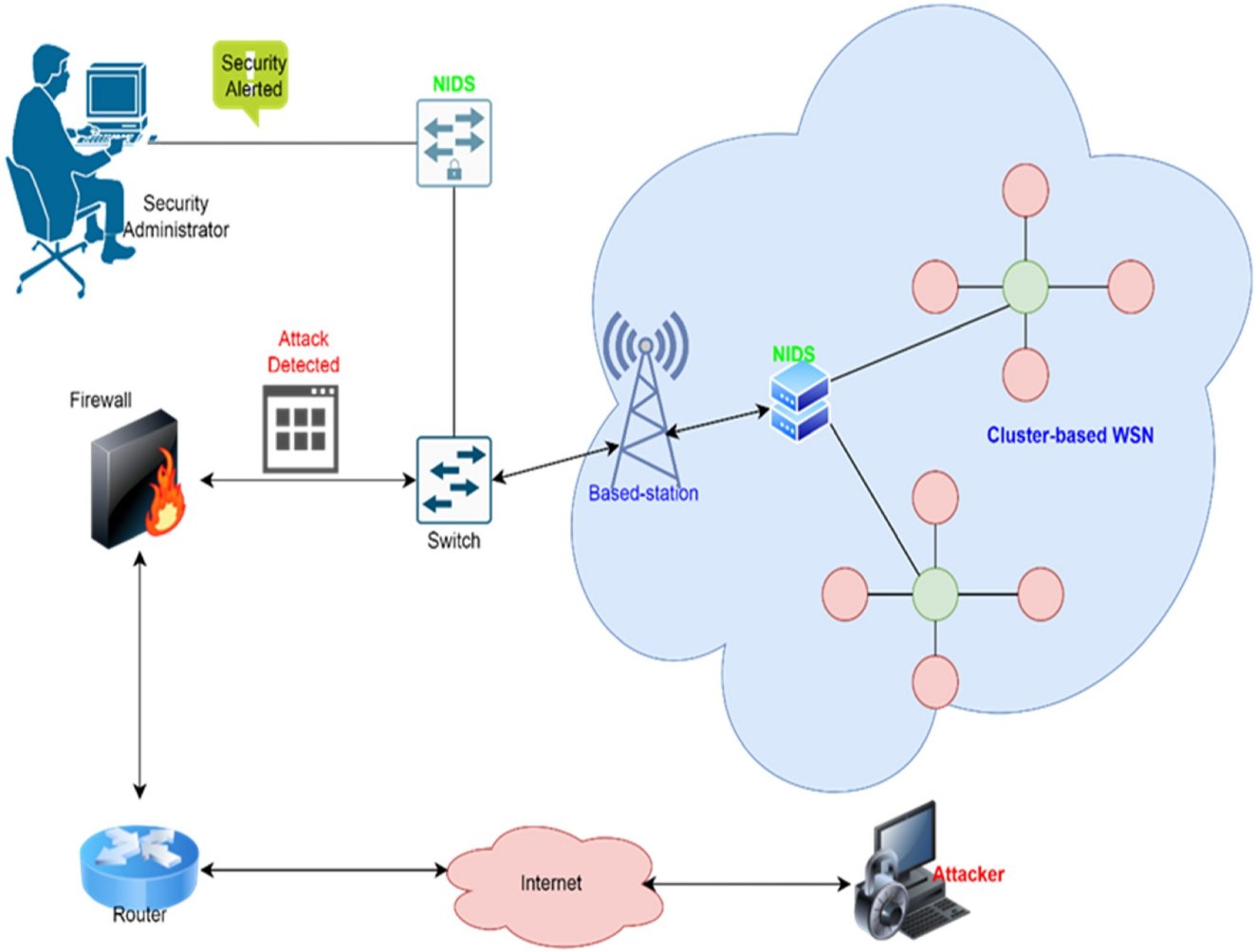

FIGURE 7 Typical Network Intrusion Detection System (NIDS) deployment strategy.

storage, has limited the operation conditions of WSNs' computational and energy resources; more specifically, the installation of security techniques such as IDS may not be directly applicable in some real-world application environments, such as military, healthcare, smart home, and so on, where sensor nodes migrate [46]. An IDS model designed for wired networks cannot be directly applied to WSNs due to their unique characteristics. In the real-world implementation of IDS in WSNs, the network structure is critical and should be considered in order to avoid a new challenge. Because a WSN necessitates a lightweight IDS, an IDS developed for the prevention of attack mechanisms in wired and ad hoc networks will not be applicable.

Though deploying IDS will aid in the detection of security risks, knowing how to respond effectively to the threats will necessitate an efficient incident response, competent security expertise, and robust procedures to resolve the situation without interfering with network operations. Intrusion detection system|intrusion detection systems security technologies are passive; they identify threats but do not act; as a result, their deployment in the real world necessitates the employment of intrusion prevention systems (IPS), which are active security devices [70]. Installing IDS sensors throughout a cluster-based WSN may not be feasible due to economic and monitoring constraints. Successfully deploying NID can be difficult, and if not done correctly, it may expose certain vital network assets. Intrusion detection system|intrusion detection systems alarms are either centrally documented using a security information and event management system or relayed to a system administrator. If not effectively handled, the high volume of alerts issued would result in high bandwidth usage and erratic network traffic flows. Investigating the alarms necessitates the use of a dedicated security specialist who is capable of interpreting the system output.

5.3 | Features selection techniques

The techniques that are used for selecting features during the pre-processing step, as well as any of the machine learning methods to develop an IDS model, are critical because they can reduce computational complexity while improving model performance. The quality of the training data, as well as the classifier used, determines the accuracy of the classification model. As a result, a pre-processing stage is routinely employed

to improve classifier performance [71]. Li and Qin [7] proposed a hierarchical and dynamic feature extraction system developed for extracting packet-level features from network traffic datasets. Still, due to the network's complexity, it was unable to detect effective feature representations in the network traffic dataset. Thus, the original flow feature distribution had to be dynamically adjusted. Dubey and Bhujade [72] presented two techniques for establishing an efficient feature subset; Kendall's correlation coefficient and mutual information were used to minimise dataset dimensionality, which improved classification performance by Using logistic regression, Nave Bayesian and One Versus The rest of the classifiers were built, but they did not perform well with layer perception. Alazzam and Sharieh [73] presented a pigeon-inspired optimiser method and a wrapper feature selection technique to improve detection accuracy and reduce training time.

A unified model of an optimised Convolutional Neural Network (CNN) with hierarchical multi-scale long short-term learning was proposed to extract and learn spatial-temporal features effectively [74], which improved detection accuracy with fewer false alarms when compared to other models. However, it had low accuracy on the NSL-KDD training dataset, which was also utilised in different models to determine the best selection of features technique. An efficient wrapper feature selection was used and evaluated to decrease the processing time of the IDS model. It was proposed that Al-Yaseen Idrees [75] use differential evolution with an extreme learning machine. Increasing the processing speed has reduced the computational overhead of the model. An ensemble classifier for dimensionality reduction was used to decide on the optimal subset based on feature correlation and to produce superior outcomes. Still, it required improved handling of rare attacks from massive network traffic [4]. Nimbalkar and Kshirsagar [75], for feature selection in machine learning IDS models, an information gain/gain ratio was utilised, which reduced the detection time while boosting accuracy.

To handle various types of threats in WSNs, For feature selection and effective classification of network trace datasets, a fuzzy rough set-based closest neighbours technique was applied [76]. Halim and Yousaf [77] proposed an improved feature selection strategy based on genetic algorithms that enhance classification precision by minimising data dimensionality, which has a negative effect on the learning algorithm's effectiveness. Herrera-Semenets and Bustio-Martínez [71] suggested a multidimensional feature selection algorithm for developing qualitative information provision that decreases the dimensionality of the IDS model's feature selection training dataset, a one-hot autoencoder was employed. Li and Chen [78] proposed additional features such as system logs and security device alarms had to be considered in order to develop an IDS model based on the random forest algorithm, which reduced detection time and significantly improved prediction accuracy, but only a few extracted from network traffic were used. Almasoudy, Al-Yaseen [79] suggested a differential evolution-based wrapper feature selection model for intrusion detection to reduce the number of features by calculating the fewest number of features without negatively impacting system efficiency, and evaluated the features by computing their accuracy with an extreme learning machine, attaining a greater detection rate while reducing false alarms.

In place of a direct consequence, a method of selecting features is a pre-processing method used in IDS development that involves locating an efficient representation of the feature attributes from the network traffic training dataset that could be utilised to increase the IDS's performance. However, by filtering features from multiple measures, the most valid features can be obtained, but this may result in overfitting issues based on Herrera-Semenets and Bustio-Martínez [71]. As a result, inefficient feature selection from the training dataset will result in poor classification, potentially resulting in biased results for any classifier algorithms built by the IDS. Often, these network traffic flow or cyber-security data have an inherent flaw: the data is highly imbalanced, involves a disproportionately large amount of typical traffic data, and, in the majority of instances, a small number of attack instances [80]. Because the majority class has a larger sample size than the minority classes, High-dimensional imbalance datasets include the NSL-KDD and UNW-NB15 datasets with high feature dimensions and large data volumes [81].

This effect may reduce detection accuracy, increase model training time, imply high computational complexity, and may also have an impact on classifier performance. Because cyber-attacks are not common, most cybersecurity datasets are imbalanced, making accurate classification of cyberattacks difficult [82]. Techniques for dealing with class imbalance include oversampling, under-sampling, and balanced sampling. By combining samples from the minority classes, oversampling was used to produce an evenly distributed dataset with a higher detection rate for those classes. By deleting some samples from the majority class, under-sampling techniques are employed to establish a more balanced distribution. This method, though, is prone to overfitting because it may omit some critical components that could be used to distinguish the majority from the minority class [83].

This problem may also cause model performance bias, resulting in low accuracy and a high number of false alarms. That is, some members of the majority or minority classes may be classified incorrectly. Thus, a model with high accuracy but a high false alarm can be a biased result if the recall score, precision score, and F-score are all low because these are the metrics used to assess class imbalance in the traffic dataset. Accuracy is simply the proportion of samples that were appropriately classified in the entirety of the samples utilised.

5.4 | Comparative analysis of intrusion detection system

A great deal of studies have been carried out on IDS in WSNs to ensure the overall network's safety even if some network parameters are compromised. [84] A bidirectional long-short-term memory IDS with efficient feature selection was proposed using Chi-square. The IDS have excellent accuracy in detecting but a high computational complexity, and they

propose that more techniques for selecting features be evaluated in order to improve performance. The deep neural network (DNN) IDS provided by Gowdhaman and Dhanapal [85] uses cross-correlation to select the optimal features and has good accuracy but decreases as the amount of malicious occurrences increases.

It is recommended that the complexity be reduced in order to improve the model's classification accuracy. Optimal support vector machine, Amaran and Mohan [86] IDS use the whale optimisation algorithm (WOA)'s kernel selection and has demonstrated good accuracy but a high false alarm rate. It was suggested that the WOA be replaced with hybrid metaheuristic algorithms to improve performance. According to Alaparthy and Morgera [87], the immune cell IDS model can detect energy-depleting attacks such as DDoS by employing a negative selection algorithm and danger theory for optimal feature selection. Han [88] built a multi-kernel extreme learning machine intrusion detection technique that decreases detection time while using little energy and has a high false alarm rate, affecting the model's overall performance. The energy trust algorithm IDS detect hybrid DoS attacks well but fails to detect other types of anomaly attacks because it uses the Pearson correlation coefficient for optimal feature selection to improve detection, a machine-learning algorithm can be used. [88].

Du and Xia [89] proposed a weighted support vector machine-optimised collaborative IDS that employs an improved artificial bee colony (ABC) for optimal feature selection, has a high detection accuracy on all anomaly attacks, but consumes a lot of energy on each node, and thus recommends a machine learning algorithm to improve the model's performance. According to Narasimha Prasad and Senthamil Selvan [90], the graph neural networks IDS developed using forward sink estimation has a good detection accuracy on sinkhole attacks. Still, it cannot be used on unstructured network layers. It is thus suggested that for improved performance, associating parameters of quick intercepting intrusions in unstructured networks be used. Alruhaily and Ibrahim [91], 2021, created a random forest multi-class IDS using a Nave Bayes classifier (NBC) for real-time decision-making, and it demonstrated high performance and precision rate on layer attacks. Still, it had a high computational complexity, which can be reduced by using deep learning.

The deep belief network model IDS with adaptive neuro Fuzzy based clustering algorithm performed better by gaining high accuracy, nevertheless at the cost-effect of increased computational complexity; further, a tuning hyperparameter method was used to regulate batch size, epoch count, and training rate to enhance model performance is recommended by Maheswari and Karthika [92]. Kurniawan and Yazid [93] created a signature-based IDS for DoS detection that showed good detection accuracy on DoS attacks yet was unable to identify other attacks, and they proposed using an evolution optimisation model as feature selection to improve model performance. A protocol layer trust-based IDS developed by Wang and Jiang [94] using networks Java enabled algorithm had high detection but created communication overhead as the hop count to the cluster-head increased, and the scheme can be optimised by analysing the attacks initiated at the layers. Ghugar and Pradhan [95] proposed an LB-IDS with a high detection accuracy on hole attack using hop count as the trust metric. They also recommend using wireless transceiver modules deployed in an outdoor setting to improve the model's efficiency. Pandey [96] presented an agent-based IDS network Java algorithm capable of self-recovering a vulnerable node after an intruder is detected. Still, the model's accuracy decreases as the number of nodes increases. To improve the overall performance of the model, he also suggests a mobile ad-hoc network and specification technique.

Martinez and Vogel-Heuser [97] proposed a WSN architecture, a host IDS to analyse data associated with specific host devices of a cyber-physical system. Godala and Vaddella [98] presented a classification of security attacks, several IDS devices for malicious detection, and Metrics for evaluating the IDS algorithm's performance in WSNs. To protect WSNs, they developed an IDS with multiple levels determined by the properties of various immune cells. They linked WSNs to white blood cells, which act as immune cells in the human body [87]. Riyad and Ahmed [99] consider the hierarchical structure of various WSN organises to be a disadvantage because a compromised node near the top of the hierarchy can bring the entire security system down, which is why they proposed a distributed IDS scenario to replace the single IDS scenario in order to detect disseminated attacks.

Elsaid and Albatati [8] presented a fortified WSN with two defence levels: Despite its adoption for speed and high scalability, if the first line of defence fails to prevent attacks, IPS and IDS are used to detect and mitigate any intrusion. Almomani and Alromi [100] state that the two main challenges of WSNs are sensor lifetime and network security. As a result, they advise using a software engineering process in the improvement of IDS in order to demonstrate its effectiveness on WSN services. Khan and Herrmann [101] conducted a detailed survey of the prevalent IDS approaches that corresponded to IoT networks, with an overview of IDSs proposed for WSNs. Colom and Gil [102], by choosing and incorporating a variety of current IDS solutions, we created a framework for appropriate scheduling of IDS tasks, but they used an IoT network with a single sensor node to deliver the performance constraints applicable to task scheduling assessments; this is the major disadvantage.

Nguyen and Phan [103] proposed a system comprised of tiered levels of intelligent IDS nodes collaborating to detect irregularities by framing policies embedded in software-defined network-based IoT access devices to stop an intruder, but only attacks based on the policy can be detected. Borkar and Patil [104] used an adaptive SVM classifier supervised machine learning technique composed of to identify suspicious activities of WSN nodes as well as for cluster head selection via network packet communication among various sensor nodes to lessen the impact of the data imbalance. Al and Dener [105] proposed a hybrid deep learning method that combined CNNs and short-term memory. The IDS model performed well when it came to detecting imbalanced datasets. Keerthika and

Shanmugapriya [106] provided a concise overview of the security problems facing WSNs; in the paper, several security attacks, such as active and passive attacks, were discussed, providing researchers with an insightful view of various types of attacks in the WSN but did not design any effective countermeasures for secured communication. At the same time, Gavel and Raghuvanshi [107] used a density valuation method based on statistical data to detect abnormal behaviour occurrences in WSNs over time in an anomaly-based intrusion detection model. Identifying anomaly activities for large dimensions of packets using the dimensionality decline method, on the other hand, continues to be a concern for the algorithm.

Mahdavi and Fanian [108] proposed an attack detection state using some code-book matrices for intrusion detection systems to hold keys of the equivalent meta-alerts as an alternative means of keeping alerts in node memory. Still, the selection process is slow, which can be improved by machine learning. Singh and Nagar [109] suggested a machine learning strategy based on a Gaussian process regression model for intruder prevention and detection, which provided a highly accurate impact in comparison to other benchmark schemes; they considered the movement of an intruder at various intrusion path angles as well as the coverage area of a mobile sensor over time. Senger [110], to reduce processing latency, a high computational capacity infrastructure environment IDS architecture was presented, with classification performed near devices and sensors.

Li and Huang [110] proposed a variational autoencoder (VAE) with Generative adversarial networks (GANs) for feature selection and uses Long Short-Term Memory (LSTM) with a multi-scale CNN as the classifier's algorithm to build an IDS. Though the performance of the model has been improved by extracting the network features at depth, the spatial inefficiency from the VAE has affected the detection accuracy. Kolukisa and Dedeturk [111] proposed a logistic regression and artificial bee colony to build an IDS model that will address attack variations. It proves to be a scalable and dynamic solution, but it has resulted in high computational complexity. Alrayes and Zakariah [112] proposed channel attention with Convolutional Neural Network to build an IDS model for anomaly detection, which has improved the performance of the model but with increased model complexity. Vibhute and Patil [113] proposed a random forest for feature selection and various classifier algorithms (Support vector machine, Logistic regression and K-nearest neighbour). However, it has increased the performance of the model with K-NN but provides less accuracy on the other classifier algorithms. Chelloug [114] proposed a CNN stacked with DNN classifier algorithms and uses a Synthetic Minority Oversampling Technique (SMOTE), which reduces the model complexity and increases the detection accuracy on novel attacks. Still, the model performance may be biased towards the minority class due to the SMOTE used.

The analysis in Table 1 below compares IDS model performance based on feature techniques, classifier algorithms used, and features selected, or the scaling feature method used, and detection accuracy evaluated on the NSL-KDD dataset, which was chosen as the basic benchmark dataset for this research work due to its popularity and the fact that it represents an appropriate proportion of normal network traffic flows with multiple attack classes. It was discovered that the feature technique used, as well as the feature scaling method used, have a substantial impact on the classifier algorithm's performance.

5.5 | Comparative analysis of intrusion detection system model using autoencoder

In order to solve the deviation in learning patterns, Ayubkhan and Yap [131] used a denoising autoencoder for feature extraction and lightGBM classifier to improve the performance of the model but also experienced a sparsity penalty problem. Gu and Wang [132] proposed a stacked sparse autoencoder for dimension reduction and memory storage in real-time response, which accelerated model detection with parallel data processing speeds but suffered from a sparsity penalty due to the one-hot-encoder used for feature extraction. De Carvalho Bertoli and Junior [133] developed an IDS using stack federated learning (FL) autoencoder ensemble and energy flow classifier, which shows effectiveness on attack detection though poor performance on some other datasets. Given a scarcity of attack datasets and only training a model on normal data, Wang and Sun [134] discovered that a one-class support vector machine with an autoencoder for feature extraction and a Gaussian mixture for anomaly detection improved detection but was still ineffective on minority classes.

Muhammad and Hossain [135] proposed an IDS for detecting transaction fraud that uses an autoencoder for feature width reduction and a DNN for feature classification, despite the fact that the model has good dimensionality reduction but can handle fewer attacks such as flooding, injection, and imitation are all examples of fraud. Ortega-Fernandez and Sestelo [136] suggested a network intrusion architecture based on a deep autoencoder for high detection of denial-of-service attacks. Still, the model was only trained on regular network flow features. Vu and Nguyen [137] proposed a conditional denoising adversarial autoencoder for sample generation and KNN. The model has high detection of DDoS, though it generated specific malicious samples. To reduce the high computational complexity in the IDS model, Cui and Zong [81] proposed a stack autoencoder for feature extraction with a Wasserstein GAN and Gaussian mixture model for class imbalanced processing, also used CNN-LSTM classifier, this has reduced the dimension and the class imbalanced but produces a low detection and low accuracy.

Kalpana [138] developed an IDS to reduce training time and model complexity using a one-hot-encoder for feature extraction, LightGBM for feature selection and recurrent non-symmetric deep autoencoder for attack classification. However, it did not identify the attack classes. To handle the emergence of signature-based attacks, DAS and PRAMOD [139] proposed a unified ensemble autoencoder for optimal feature selection but could not identify SQL injection and botnet attacks.

TABLE 1 The analysis of intrusion detection system (IDS) Model Comparison on NSL-KDD Benchmark dataset.

References	Feature selection techniques	Classifiers	Features selected	Advantages	Disadvantages	Future directions	Accuracy (%)
Kanna and Santhi [115]	ABC	BWO-Conv-LSTM	30	Have reduced the rate of model complexity.	Not able to detect high similarity in attack class.	Other public network intrusion datasets must be adapted.	98.67
Albahar, Binsawad [116]	L1&L2 Norm	ANN	2	Detect any irregularities in data transmission.	It has resulted in model complexity.	Find a possible way of reducing the model complexity.	98.5
Elsaid and Albatati [8]	IABC	WSVM	Optimal	Detect all classes of anomaly attacks.	The complexity of the model has increased.	An algorithm could be used to improve performance.	98.4
Benmessahel, Xie [117]	ANN	MVO	2	Identified several classes of attacks.	Did not used adequate feature selection technique.	Use a technique that can minimise the number of selected features.	98.21
Zhang, Li [118]	DBN	IGA	2	Reduced model complexity.	Has high training time.	Optimises the model to reduce the training time.	98.07
Hajimirzaei and Navimipour [119]	ANN	ABC	4	Discriminate between normal & abnormal packets flow.	No detection has been done.	Improve the model for attacks detection.	96.81
Safaldin, Otair [120]	IGWO	SVM	29	Speed up processing time.	Has reduced the classification process.	Could improve using different classifier algorithm.	96
Gowdhaman and Dhanapal [85]	CC	DNN	Optimal	Has good accuracy at an instance.	The accuracy reduces with increased number of attacks.	Could increases the classification accuracy.	95.53
Zhou, Cheng [4]	CFS-BA	RF	17	Able to detect various form of attacks.	Produced high false alarm on some rare attacks.	Improve and respond to rare network traffic attacks	94.9
Dubey and Bhujade [72]	Dense_FR & Sparse_FR	Hamming loss & Jaccard score (NB)	20 7	Has reduced the training time.	Have low detection on anomaly attacks.	Need to improve the performance classification.	94.59 94.96
Amaran and Mohan [86]	WOA	OSVM	Optimal	Having a good accuracy.	Produces high false alarm.	Could improve with algorithm.	94.09
Benmessahel, Xie [121]	ANN	LSO	2	Has used an adequate feature selection.	Have low detection.	Improve the overall performance.	94.02
Benaddi, Ibrahimi [122]	PCA-fuzzy	KNN	3	Good dimensionality reduction.	Has result in linear correlation issue on the dataset.	Used technique that can improve the feature selection to improve the detection.	94
Vinayakumar, Alazab [123]	L2 Norm	DNN	2	Classified unforeseen (unpredicted attacks).	It was not trained to used NIDS dataset.	Could add module for monitoring network event & not just host events.	93.4
Zhang, Han [124]	MK-function	MK-ELM	2	Reduced detection time.	Produced high false alarm rate.	Improve the overall performance	92.10
Kanna and Santhi [74]	OCNN	HMLSTM	Optimal	Automatic learning of both spatial & temporal features.	Has low detection rate.	Improve to have a high detection on traffic datasets.	90.67
Tang, Gu [125]	OHE	DSN	2	Used several machine learning.	Has leaded to overfitting problem.	May improve using deep learning with automatic dimensionality reduction.	90.41
Alazzam, Sharieh [73]	Cosine_PIO	DT	5	Achieved low false alarms.	Have low detection accuracy as the result of discrete optimization issue.	Discretisation process with swarm intelligent algorithm may resolve the problem.	88.3

TABLE 1 (Continued)

References	Feature selection techniques	Classifiers	Features selected	Advantages	Disadvantages	Future directions	Accuracy (%)
Al-Yaseen, Idrees [126]	DE	ELM	3	Improved processing time.	Could not detect U2R attacks & have low detection on R2L some type of attacks.	Could use more complex classifier to enhance the detection.	87.7
Almasoudy, Al-Yaseen [79]	DE	ANN	5	The system's performance was not affected by using the fewest features possible.	Could not detect R2L attacks.	Could use more complex classifier to enhance the detection.	87.53
Li, Chen [78]	RF	AE-NN	Optimal	Does not depend on labelled training dataset.	Have low detection due to overfitting.	May improve using spatially efficient autoencoder.	85.0
Gao, Shan [127]	VC	MultiTree	17	Gather the advantages of several algorithms.	Could not detect U2R attacks & delay in detection.	Optimisation method should be considered for model improvement.	84.23
Wu, Chen [128]	CNN	CNN	11	Achieved low false alarm rate.	Have low detection rate.	Modifying the structure of the CNN model; may improve the model.	83.2
Imrana, Xiang [84]	X^2	Bit-LSTM	17	High intrusion detection accuracy.	Have a high computational complexity.	Exploring more feature selection algorithms to improve the model.	82.05
Ahsan, Shi [82]	OHE,	FCNN,	32	Used different sampling method.	Have an overfitting & spatial issue.	May improve using spatially efficient autoencoder.	82.0,
	SMOTE,	VC,	32				83.0,
	US,	VC,	32				82.0,
	SMOTEEN	VC	32				82.0
Ding and Zhai [129]	OHE	CNN	5	Reduced training time.	Have high false alarm, & low detection for U2R & R2L attacks.	May improve using spatially efficient autoencoder.	80.13
Bedi, Gupta [130]	Siamese NN	B-XGBoost	2	Reduced computational complexity.	Have an overfitting that led to low detection accuracy.	May improve using spatially efficient autoencoder.	80.1
Li, Huang [110]	VAE + GAN	LSTM + MSCNN	Optimal	Extract network features at depth	Spatial inefficiency from VAE affects the accuracy	Improve the spatial efficiency of the model	83.45
Kolukisa, Dedeturk [111]	LR	ABC	Optimal	Provides scalable & dynamic solution	High computational complexity	Reduces the model complexity	90.11
Alrayes, Zakariah [112]	CA	CNN	Optimal	Improves performance	High computational complexity	Reduces the model complexity	99.73
Vibhute, Patil [113]	RF	SVM, LR, & K-NN	Optimal	Improves performance	Less accuracy on LR & SVM	Consider deep learning	98.24
Chelloug [114]	SMOTE	CNN-DNN	Optimal	Reduced model complexity & increased performance	May be bias towards the minority class by SMOTE	Use other feature selection method	99.0

Alissa and Alotaibi [140] use a modified deer hunting optimisation-based feature selection to select features and crystal structure optimisation with a deep autoencoder as a classifier to develop an IDS with high detection of attack classes for drone privacy risk. Still, she has low detection of the R2L class of attacks. Khanam and Ahmedy [141] developed an IDS using class-wise focal loss for generating a minority sample, VAE for dimensional reduction, and with DNN as the classifier, though has achieved a balanced dataset but resulted in low detection and model overfitting. A stochastic fractal search algorithm with a deep learning-driven IDS model was proposed by Duhayyim and Alissa [81], using chicken swarm optimisation of deep stack autoencoder for high attack detection. Still, it was unable to detect an outlier attack. In

order to detect an advanced persistent threats, Neuschmied [142] proposed an IDS using Principal component analysis (PCA) for dimensionality reduction and coupled with autoencoder to identify the anomaly, but lack comprehensive datasets for such attacks.

Li and Chen [143] propose an IDS using Denoise autoencoder for data augmentation and GAN to generate and extract spatial features from traffic flows, which has a low false alarm as the result of random sampling in the data space but led to low detection recall in the model. Wang and Du [144] developed an IDS with low dimensional features using a stack contractive autoencoder for feature extraction and support vector machine classifier though it required high training time. In order to handle a polymorphic threat, the authors in ref. [145] propose an IDS using one-dimensional CNN as a classifier and deep stack autoencoder for detection. Though it has good performance and low false alarms, it is not sensitive to some attack classes, such as DDoS. He and Wang [146] used conditional Wasserstein VAE with the GAN for feature extraction by generating a minority class sample and using a one-dimensional CNN for attack detection, the model has an improved performance but low detection.

Sumathi and Rajesh [147] propose an IDS model using LSTM autoencoder for feature extraction and Harris Hawks optimisation with particle swarm optimisation to select features and classify the attacks. However, it selected an optimal feature that yielded better performance but was not able to detect new DDoS attack instances. Ketepalli and Bulla [148] proposed an IDS model using LSTM-AE for feature selection and Random Forest for attack detection, which reduced the computational complexity of the model but had low detection of R2L and U2R classes of attacks. In order to handle Overfitting and class imbalance, Chikkalwar and Garapati [149] use an autoencoder to create instances of minority classes, and grasshopper optimises SVM for attack detection, which improves performance but does not identify attack classes. Wang and Liu [150] proposed an IDS model with an autoencoder for feature extraction, PCA for further dimension reduction, and KNN, Decision Tree, AdaBoost, and Bagging classifiers, though it has efficient feature selection, is effective on KNN, and has poor accuracy with Decision Tree, AdaBoost, and Bagging.

Mhamdi and Isa [151] presented a hybrid Deep Autoencoder-Random Forest model to solve numerous security concerns in Software-Defined Networking, such as unauthorised access and DoS attacks. Despite producing a relatively high detection rate and a low false-positive rate, combined with minor performance overhead on network controllers, it was unable to address the spatial aspects required for an effective IDS. Le and Truong [152] consider spatial features for a robust IDS proposed a combination of Time-Embedded Transformer and Autoencoder (AE) to solve vulnerabilities in Controller Area Network communication. This combination allows the system to capture packet and sequence-level features, which improves its ability to detect complex and real-time threats. Bi and Guan [153] propose a hybrid method that combines several advanced techniques: stacked sparse contractive autoencoders for feature extraction, attention-based bidirectional long-short-term memory (LSTM) for classification, and a Decision Fusion algorithm to improve the final classification outcome. Though it enhanced the IDS's accuracy and efficiency, the model's resilience can be increased by addressing temporal-spatial aspects.

Khaw and Jahromi [154] present a novel iterative-based method for creating adversarial samples that can bypass autoencoder-based cyberattack detection systems, demonstrating the sophistication required to bypass Machine Learning-based systems. However, it also reveals a critical vulnerability in autoencoder-based anomaly detection systems, emphasising the importance of evaluating the robustness of these systems before deployment in the real world. Tahir and Abdullah [155] developed a Deep Learning-Based Missing Data Imputation (DMDI) IDS model that uses a stacked denoising autoencoder and Gradient Boosting to increase imputation accuracy. This methodology improves overall performance by efficiently handling missing data across five classifiers (SVM, KNN, Logistic Regression, Decision Tree, and Random Forest). Still, it does not investigate the method's application to other domains with concerns about missing data. Hore and Ghadermazi [156] propose an AI-based multistage detection framework that combines DNNs with transfer learning techniques to improve the identification of known and unknown network threats, including zero-day and adversarial attacks. While it handles a wide range of attack types, it generates a high false positive rate and computational complexity.

Nixon and Sedky [157] suggested a method, Split Active Learning Anomaly Detector, which uses autoencoders for anomaly detection and combines them with a split active learning framework to reduce labelling costs. It has much shorter processing times, making it more efficient for real-time anomaly detection. Shrestha and Mohammadi [158] recommended using LSTM and autoencoders in conjunction with FL techniques to improve data privacy and cybersecurity. Though using an LSTM-autoencoder for time-series data anomaly detection ensures robustness in detecting anomalous behaviours, relying on synthetic datasets may limit real-world application.

Long and Xiao [159] use recursive methods for optimal feature selection and autoencoder ensembles for the detection of the proposed IDS model. This method has reduced the training time but produces low detection accuracy on some attack classes. Kumar [147] found an ideal topology set of CNN using an evolutionary algorithm in an IDS model, using a multichannel autoencoder for feature selection and CNN for attack detection. Still, the accuracy of most classifiers and datasets used is poor. In order to handle the autoencoder sparsity problem, Rao and Rao [160] employ a sparse autoencoder with smoothed II regularisation for feature selection and a DNN classifier for attack class multiclassification, resulting in weak detection of R2L and U2R attack classes. Binbusayyis and Vaiyapuri [161], to classify the attack, suggest an IDS model that uses compact feature selection with a one-dimensional convolutional autoencoder and a one-class

support vector machine. It detects novel attacks but does not identify attack classes. The analysis of the IDS based on autoencoder is shown in Table 2.

5.6 | Description of the methodologies used in intrusion detection system

The filter, wrapper, and embedded techniques are some of the most common feature techniques used in IDS models [164]. The filter technique is a statistical method used to analyse the relationship between input and target variables. It is applied directly to the data by computing the threshold value, determining whether the feature will be selected or rejected. While the filter technique seems less expensive and reduces the complexity of a model by making it easier to interpret, when there is little redundancy in the data, it performs poorly [165]. The wrapper technique creates a subset of features from the whole feature set using forward selection, backward elimination, or recursive feature elimination iteration. Then, they are used to train a model [166]. It outperforms the filter technique by selecting an optimal feature subset that can improve system performance and accuracy. Still, its main disadvantage is that it requires a lot of computational power and memory [167] and results in overfitting when the amount of data is insufficient [168].

The embedded method combines the benefits of the filter and wrapper methods and implements them using algorithms with built-in feature selection methods [169]. It is less prone to over-fitting and outperforms the filter and wrapper techniques because it learns the best feature subset while developing the model, making it faster than the previous techniques [170]. However, it has several challenges, depending on the algorithm selected. Another method used is oversampling, under-sampling, and hybridisation [71, 105, 126]. Oversampling techniques, such as the SMOTE, were used to create a balanced dataset sample with a high detection rate for the minority classes, and under-sampling techniques were used to achieve a balanced distribution of minority and majority classes, whereas hybrid techniques combined oversampling and under-sampling techniques to improve performance.

These techniques, however, are prone to overfitting because they may omit some critical components or feature attributes that could be used to distinguish the majority from minority classes [83]. Overfitting is a problem that occurs when a model becomes too closely tailored to the specific data in the dataset. It can result in the model performing better on the training dataset than on the test dataset. Overfitting can also occur when the model is too complex due to high dimensionality and class imbalance [81, 82]. As a result, biased model performance may result; in this case, the system may correctly classify the sample dataset but have low detection accuracy for anomaly attacks.

Principal component analysis is another effective feature selection technique utilised by many researchers to cope with the dimensionality issue in network traffic datasets during feature selection, and it is effective in anomaly detection [122]. Principal component analysis is a valuable tool for feature selection in intrusion detection systems. However, it is a linear transformation that may not capture non-linear correlations between feature attributes. It is a problem because most features in network traffic datasets are non-linearly correlated. As a result, using PCA for feature selection may result in a high number of false alarms [171–174]. However, it was found that autoencoders outperformed all other feature selection techniques most researchers utilise for dimensionality reduction [161, 175–179].

The literature review analysis, as shown in Tables 1 and 2, shows that researchers used various techniques to improve the performance of the IDS models. No feature selection process, however, outperformed all classifier algorithms equally well. Several machine learning approaches have been used in IDS development by various researchers, such as the traditional machine learning algorithms (NBC, DT, SVM, KNN, LR and so on), ensemble learning algorithms (AdaBoost, RF, Bootstrapping and so on), and deep learning algorithms (CNN, LSTM, RNN, DBN, AE-DLA and so on). Others have attempted to improve the performance of these algorithms to obtain an optimised model. Furthermore, according to the analysis, different feature selection techniques and classifier algorithms used by various researchers lead to their essential performance, which can also be deduced as neural network models outperforming most of the approaches used.

Autoencoders outperformed each other in dimensionality reduction and class imbalance problems on network traffic datasets for feature selection [161, 175–179]. However, not all of the features in a network traffic dataset are required to build an efficient IDS model, and this necessitates the use of an essential features selection technique that can effectively and efficiently represent the feature attributes in the datasets to reduce the model's training time and improve its accuracy. Furthermore, when choosing an efficient feature representation as shown in Tables 1 and 2, it is necessary to consider the relevance of the features as well as their effectiveness on the classifier algorithm and training datasets first.

Figure 8 depicts the various feature selection techniques used by most researchers in developing the IDS model and their strengths and limitations. Some studies used filter techniques, which are less expensive and easier to grasp but perform poorly on datasets with little redundancy [168, 180–182]. To address this weakness, a wrapper strategy was devised that selects an appropriate feature subset from the dataset via forward selection or backward elimination; however, it has significant computing complexity and leads to model overfitting [183–185]. It was addressed by an embedded technique with a built-in feature selection algorithm, which is less prone to overfitting but presents several challenges depending on the algorithm used [186–189].

An oversampling technique was applied by synthesising the attack classes in the dataset to balance it with the amount of regular traffic to address the issues posed by this technique. However, it is prone to overfitting [190–192]. Under-sampling techniques solved the problem by discarding some of the normal traffic's features and balancing them with the attack

TABLE 2 Analysis of intrusion detection system (IDS) based on autoencoder.

References	Identified problems	Proposed solutions	Used datasets	Strengths	Weaknesses
Gu, Wang [132]	Limited storage	SSAE for dimensionality reduction & SGRU for real-time response	AWID	Parallel data process speeds up model detection	Sparsity penalty from OHE for feature extraction
Ayubkhan, Yap [131]	Deviation in learning patterns	Denoising AE for feature extraction & light GBM for classification	CIDDS-001, CIDDS-002, ISCX-URL2016, UNSW-NB15, CIC-IDS-2017, ISCX-Tor2016, BoT-IoT, IoTID20 and Kyoto 2006+	Removed noise & improved performance	Sparsity penalty from OHE for feature extraction
De Carvalho Bertoli, Junior [133]	Poor performance of dataset to other networks	Stack federated learning AE ensemble & energy-flow classifier	TON_IoT, Bot-IoT, CICIDS2018, UNSW-NB15	Improved attack detection on Bot-IoT & TON-IoT datasets	Poor performance on other datasets
Wang, Sun [134]	Lack of enough attack data at hand	One class SVM with AE for feature extraction & Gaussian mixture for anomaly detection	NF-BoT-IoT-V2 & NF-CSE-CIC-IDS2018-V2	Improved detection	Train model only with normal data, ineffective anomaly detection
Muhammad, Hossain [135]	Transaction fraud	AE for feature width reduction & DNN for feature classification	KDD Cup99, NSL-KDD, & AWID	Good dimensionality reduction	Handles fewer attacks; flooding, injection & impersonation
Ortega-Fernandez, Sestelo [136]	Denial of service attacks	NID architecture based on a deep AE	Cyber security ICS dataset captured in real-time	High detection of DDoS attacks	Train model on normal & network flow features only
Vu, Nguyen [137]	Class imbalance	Conditional denoising adversarial AE for sample generation & KNN for borderline sample	Augmented dataset	High detection of DDoS	Generate specific malicious samples
Cui, Zong [81]	High computational complexity	Stack AE for feature extraction, GMM-WGAN for class imbalance processing & CNN-LSTM for classification	NSL-KDD & UNSW-NB15	Reduced dimension & class imbalance	Low detection & low accuracy
Kalpana [138]	Non-feasibility of traditional approaches	OHE for feature extraction, LightGBM for feature selection, & RNDAE for classification	NSL-KDD, CICIDS2017, & CSECICIDS2018	Reduced training time & model complexity	Detection is not on attack classes
DAS and PRAMOD [139]	Emergence threats	Unified ensemble AE for optimal feature selection	CICIDS18	High detection for signature-based attack	Could not identify SQL injection & botnet attacks
Alissa, Alotaibi [140]	Drone privacy risk	MDHOFS for feature selection & CSODAE for attack detection/ classification	NSL-KDD	High detection of attack classes	Low detection of R2L class of attacks
Khanam, Ahmedy [141]	Imbalance data	CFL generate a minority sample, VAE reduce the dimension, & DNN for classification	NSL-KDD	Balanced the dataset used	Low detection accuracy as the result of overfitting
Duhayyim, Alissa [162]	Data pollution attacks	SFSA for feature selection, CSO optimised DS-AE for detection/ classification	NSL-KDD, & CICIDS2017	Good attack detection	Outlier attack detection problem
Neuschmied, Winter [142]	Advance-persistent threats	PCA for dimensionality reduction, & AE for anomaly detection	CICIDS2017	High detection of anomaly attacks	Lack of comprehensive datasets for APT attacks
Li, Chen [143]	High false alarm	DAE for data augmentation, & GAN generate extract spatial features from traffic flows	NSL-KDD, & UNSW-NB15	Low false alarm as the result of random sampling in data space	Low detection recall
Wang, Du [144]	High dimensionality	SCAE for feature extraction & SVM for classification	KDD Cup99 & NSL-KDD	Obtained low-dimensional features	Required high training time
Badji and Diallo [145]	Polymorphic threats	One-dimensional CNN & DAE for anomaly detection	CSE-CIC-IDS2018	Low false alarm	Is not sensitive to some attacks like DDoS
	Class imbalance		NSL-KDD, & UNSWNB15		Low detection

TABLE 2 (Continued)

References	Identified problems	Proposed solutions	Used datasets	Strengths	Weaknesses
He, Wang [146]		CWVAEGAN for feature extraction by generating minority class sample, & IDCNN for attack detection		Achieved class-balanced dataset	
Sumathi, Rajesh [147]	Delay convergence of DDoS attack detection	LSTM-AE for feature extraction, & HHO-PSO for feature selection & classification	NSL-KDD	Optimal feature selection yielded better performance	Unable to detect new DDoS attack instances
Ketepalli and Bulla [148]	High computational cost	LSTM-AE for feature selection, & RF for attack detection	NSL-KDD	Reduced computational complexity	Low detection of R2L & U2R classes of attacks
Chikkalwar and Garapati [149]	Overfitting & class imbalance	AE generate instances of minority classes; GO optimises SVM for attack detection.	UNSW-NB15, CICIDS2017, NSL-KDD, & Kyoto2006+	Improved performance in attack detection	Did not identify attack classes
Wang, Liu [150]	High-dimensional & overfitting	AE for feature extraction, PCA for further dimension reduction, & KNN, DT, AdaBoost & bagging classifier	ICS dataset (tommy-Morris-uah)	Efficient feature selection & effective on KNN	Low accuracy with DT, AdaBoost & bagging
Long, Xiao [159]	High dimension & class imbalance	Recursive for optimal feature selection & AE ensembles for detection	NSL-KDD, CSE-CIC-IDS2017, & UNSW-NB15	Reduced training time	Low detection accuracy on some attack classes
Pandey, Kumar [163]	High false alarm & low detection	Multichannel AE for feature selection & CNN for attack detection	KDD Cup99, UNSW-NB15, & CICIDS2017	Discover the ideal topology set of CNN using an evolutionary algorithm	Low accuracy on most classifiers & datasets
Rao, Rao [160]	AE sparsity problem	Sparse AE with smoothed II regularisation for feature selection & DNN classifier.	KDD Cup99, NSL-KDD, & UNSW-NB15	Multiclassification of attack class	Poor detection of R2L & U2R attack
Binbusayyis and Vaiyapuri [161]	Class imbalance	Compact feature selection with 1D-CAE, & OCSVM classifier	NSL-KDD, & UNSW-NB15	Detect novel attacks	Does not identify attack classes
Mhamdi and Isa [151]	DoS attack	Deep autoencoder & random forest	CICIDS2017	High accuracy & low false alarm	Unable to address the spatial effect on the features
Le, Truong [152]	Spatial effect on temporal features	Time-embedded transformer & autoencoder	ROAD dataset	Improves the ability to detect complex & real-time threats	Limited to synthetic data
Bi, Guan [153]	Low classification accuracy	Stacked sparse autoencoder & LSTM	UNSW-NB15	Enhanced accuracy & efficiency	Fail to address temporal spatial feature
Khaw, Jahromi [154]	Adversarial attacks	Iterative-based autoencoder	MITM-FDI	Reveal critical vulnerability of autoencoder	The white-box assumption limits its real-world application.
Tahir, Abdullah [155]	Missing data imputation	Stacked denoising autoencoder with gradient boosting	NSL-KDD & UNSW-NB15	Enhances the overall performance	Does not explore other domain with missing data problem
Hore, Ghadermazi [156]	Zero-day attack & adversarial attack	AI-based, DNN with transfer learning	CICIDS2017 & CICIDS2018	Handles a variety of advance attacks type	Generate false positive & high computational complexity

(Continues)

TABLE 2 (Continued)

References	Identified problems	Proposed solutions	Used datasets	Strengths	Weaknesses
Nixon, Sedky [157]	Computational complexity	Split active learning anomaly detector & autoencoder	KDD Cup 1999 & UNSW-NB15	Reduces labelling cost while detecting attacks	Does not detect adversarial attack
Shrestha, Mohammadi [158]	Data privacy & cybersecurity	LSTM & autoencoder with federated learning	Synthetic industrial dataset	Ensures robustness on attack detection	Synthetic data limits the real-world applicability

FIGURE 8 An overview of the research gap.

class; however, this resulted in under-fittings, which were addressed by hybrid techniques but increased model complexity due to the high dimensionality effect [191, 193–195]. Principal component analysis was developed to address the dimensionality issue; however, while it is an effective linear dataset transformation, it may fail to capture a non-linear relationship between feature attributes [196, 197]. An autoencoder has been shown to perform better in dimensionality reduction and class imbalance by having an automated feature selection for linear and non-linear feature correlations [198–200]. Still, it is difficult to detect multiple attacks in different locations of the network traffic flows, which is known as spatial inefficiency and has yet to be addressed by any researcher.

5.7 | The benchmark dataset

An attack dataset reflects the real-world attack scenarios from the laboratory's simulated cyberattack experiments [201]. In the experimental evaluation, four separate datasets; CICDS-2018, UNSW-NB15, WSN-DS and NSL-KDD datasets were employed:

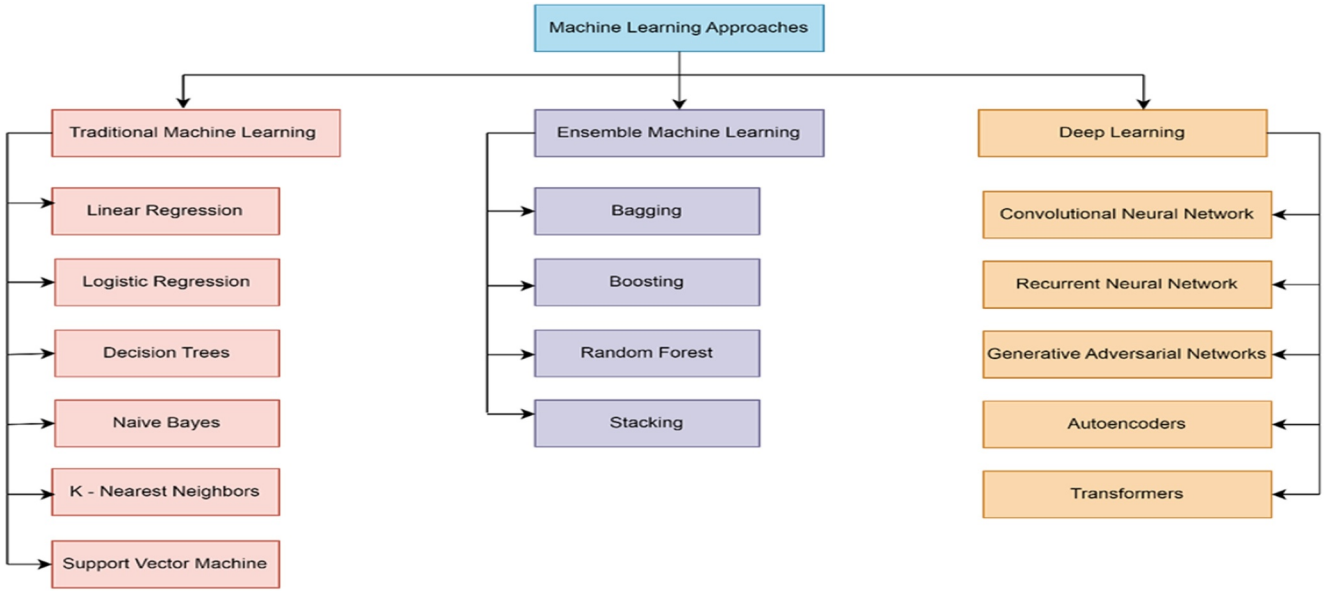

FIGURE 9 Classification of machine learning approaches used in intrusion detection system (IDS).

The Communications Security Establishment and the Canadian Institute for Cybersecurity cooperated to create the CICIDS-2018 dataset [202]. It had 80 features initially, including six attack categories and one Normal. It is a comprehensive benchmark dataset commonly used for IDS evaluation, and it includes a variety of network traffic instances for simulating real-world network environments.

The UNSW-NB15 dataset contains 45 features [203], three of which are character categorical (proto, service, and state) and 10 attack class labels: DoS, worms, exploits, analysis, generic, shellcode, reconnaissance, fuzzers, backdoors, and Normal. The UNSW-NB15-NB15 dataset was created with IXIA Perfect Storm by the Australian Centre for Cybersecurity, and it is a network-based dataset that captures modern traffic patterns and low-footprint intrusions [204, 205].

The WSN-DS is a specialised dataset for detecting four types of DoS attacks in a WSN, specifically CBWSN: black-hole, flooding, grey-hole, and scheduling attacks, all of which are referred to as energy depletion attacks. Almomani and Al-Kasasbeh [206] created the dataset in a Network Simulation Two (NS2) environment with 100 nodes in a 10,000-square-metre region. It resulted in 18 attributes of a class label of around 374,661 data records for intrusion detection systems in WSNs. The dataset can be used to prevent infiltration by prohibiting malicious nodes from entering the network, with DoS attacks being the most hazardous and damaging on WSNs due to vulnerabilities to security threats.

The most commonly used dataset for analysing network internet traffic is the NSL-KDD dataset, and the KDD Cup was a 1999 international knowledge discovery and data mining tools competition to gather traffic data [84]. The competition aimed to develop a network intrusion detection model that can be used to differentiate malicious network connections from Normal traffic. As more than just a direct consequence, a large volume of internet traffic data was collected and bundled into the KDD-99 data set, and the NSL-KDD was brought in from the University of New Brunswick as the cleaned-up version [207]. The dataset contains four types of attacks that an anomaly IDS can detect: DoS, Probe, U2R, and R2L. When the network system is used, it is targeted by two types of attackers: authorised users and legitimate users who have exceeded their legitimate boundaries. The second type is an illegal user, who attempts to gain unauthorised access to the network system to operate or attack critical network components [208]. Attacks on WSNs can be either an active attack aimed at destroying network assets or a passive attack aimed at stealing valuable information from the networks.

6 | APPROACHES OF MACHINE LEARNING FOR INTRUSION DETECTION SYSTEMS

There are three types of machine learning approaches for developing intrusion detection systems as shown in Figure 9 below: traditional machine learning, ensemble learning algorithms, and deep learning algorithms. In IDS development, machine learning algorithms are used mainly for model classification. The process of identifying, comprehending, and grouping data based on certain common behaviour is known as classification. The training dataset is classified in machine learning based on its categories. The algorithm predicts whether it is an attack or a normal dataset. The prediction's accuracy is determined by the training dataset's quality and the feature selection technique, which determines whether the selected feature tends to represent the entire feature in the original dataset without bias.

6.1 | Traditional machine learning algorithms

Traditional machine learning algorithms are a collection of algorithms that have long been used and studied in the field of machine learning [208]. These algorithms are based on mathematical and statistical models, and they have been thoroughly tested and evaluated on a wide range of tasks and datasets. These algorithms have been extensively researched and optimised over time and have a long history of use in various applications [209]. They are straightforward, simple, and have well-established theoretical foundations. They are also less computationally expensive and easier to implement than deep learning and other more recent machine learning algorithms [210]. Traditional machine learning algorithms are not always the best choice for every problem, and they may not be able to handle more complex and non-linear data relationships. However, they are still widely used in many applications and can be a good starting point for many problems [211]. Traditional machine learning algorithms that are commonly used include the following:

i. *Linear Regression* is a method of modelling the relationship between one or more independent variables and a dependent variable. A continuous target variable is predicted using linear regression [212, 213].
ii. *Logistic regression* is a statistical technique for analysing a dataset in which one or more independent variables influence the outcome. A binary target variable is predicted using logistic regression [214]. It is used to generate a binary prediction as a result of an analysis of a dependent variable in order to develop an independent variable that can be categorical or numerical [215]. Still, the dependent variable is always of the unconditional variety. A linear model of classification can be utilised to classify a binary output for the logistic function, commonly known as the sigmoid function, which is used in predictive analysis [216]. It determines the chance of belonging to one of the output classes ranging from 0 to 1. It is simple to implement and takes low processing resources; nonetheless, it may not be ideal for non-linear scenarios [217].
iii. *Decision trees* are a tree-based model; each network node represents a feature, each branch represents a decision, and each leaf node represents the result. Both classification and regression tasks are performed using DT [218]. Its branches are used to categories several data sets into separate categories. The decision tree algorithm can instantly eliminate features that are unnecessary or irrelevant. Learning includes the processes of feature selection, tree formation, and pruning [169]. When training a decision tree model, the algorithm selects the best features and produces child nodes from the root node [219].
iv. *Naive Bayes:* It is a supervised technique of classification that employs the Bayes rule and models the class-conditional variance in each attribute separately [220]. Although it is considered to be time-effective, it conforms to the "naive" assumption of independence between any given input feature combination [221]. It is a Bayesian-based classification method using probabilistic techniques. The term "naive" is used to indicate that it tends to make a strong assumption of independence between the features [222]. It can be utilised to determine the likelihood of data falling within a particular dataset category.
v. *K-Nearest Neighbours:* is a method for categorisation that is not parametric and regression. The algorithm finds the k-nearest neighbours of a given data point and classifies it based on the majority class of its k-nearest neighbours. [223]. It is incorporated into a training dataset for pattern recognition to train the classifier based on its nearest neighbours. As a result, the classification outcome is limited to the top-k nearest neighbours [224]. The parameter k has a significant impact on the accuracy of KNN models. The lower the value of k, the more complex the model is and the greater the risk of overfitting. The larger k, on the other hand, the simpler the model and the weaker the fitting ability.
vi. *Support Vector Machine (SVM):* is an algorithm for supervised learning that can perform both regression and classification tasks [225]. In the feature space, the algorithm identifies the optimum higher-dimensional space for separating distinct classes [226]. This technique is employed to train and classify data that has a specific degree of polarity. SVMs can yield satisfactory results, but considering that the disconnection feature space is established by a small number of support variable vectors, small-scale training sets are appropriate [227]. SVMs, on the contrary, are in the same direction, sensitive to noise around the hyperplane and are excellent at solving linear issues; kernel functions are often employed for non-linear data. A kernel function transforms the original space into a new space, allowing non-linear data to be discriminated; SVMs and other machine learning algorithms typically use kernel methods [228].

6.2 | Ensemble learning algorithms

Ensemble learning is a machine learning approach that maximises performance by integrating multiple basic classifier models [229]. It is a machine learning method that incorporates various models to actually create more accurate and stable predictions than any single model could produce [68]. It is based on the idea that by combining the predictions of multiple models, the ensemble can leverage each model's strengths while overcoming their weaknesses [230]. Ensemble learning can be a powerful technique for improving machine learning model performance. It reduces overfitting by averaging the predictions of multiple models, and it also increases model diversity, which can lead to better generalisation performance [229].

It should be noted; however, that ensemble learning is computationally costly and demands a large amount of data to train efficiently [231]. It is also essential to keep in mind that ensemble learning may not be the best option for every

problem, and the performance of it is necessary to evaluate the ensemble to the accomplishments of the individual models to determine if it is an appropriate technique for a given problem. The common uses of ensemble learning are as follows:

i. *Bagging* is an abbreviation for "bootstrap aggregation" [232]; it is a technique that involves training multiple instances of the same model on the training data. It is divided into random subsets, and the predictions are combined by a majority vote or by averaging the predictions [233].
ii. *Boosting* is a technique in which multiple instances of the same model are trained with the same training data but at different weights to the training examples [234]. At each iteration, the weights are updated so that the model focuses more on the examples that were misclassified in the previous iteration [235].
iii. *Random Forest:* A bagging extension that decorates base models by training them on different random subsets of features [236]. It consists of multiple DT built into a training dataset. New data is fit as a random forest under any of the trees rather than being forced to be an essential data point within only one category, as is the case with DT [237].
iv. *Stacking* is a technique that involves training multiple models on the same training data and using their predictions as features for a higher-level model that makes the final prediction [238].

6.3 | Deep learning algorithms

Deep learning is a machine learning subfield that uses multilayer artificial neural networks, also known as DNNs [239]. Relatively high features are derived from lower-level features in these networks, which are designed to learn hierarchical data representations [240]. Deep learning algorithms can now learn a hierarchy of features from raw data, which is particularly useful for image and speech recognition tasks [241]. Deep learning algorithms have achieved dynamic performance in speech, image recognition, natural language processing, and gameplay, which are just a few of the tasks that can be performed [242]. They can handle large and complex datasets and can learn features from raw data automatically. They are, however, computationally expensive to train, necessitating large amounts of data and powerful hardware [243]. It is also worth noting that deep learning is not always the best choice for every problem, and it is vital to contrast the outcomes of deep learning to the performance of traditional machine learning algorithms to determine if it is an appropriate technique for a given problem. The following are the commonly used deep neural networks:

i. *CNNs:* These are neural networks that excel at image recognition tasks [244]. They are made up of several layers, including convolutional layers for extracting features from images and fully connected layers for classifying images, as shown in Figure 10. A CNN was first developed, and It has been demonstrated that using a model to map images to outputs is useful in any prediction circumstance [221]. Its hidden layers usually consist of convolutional and combining networks and a fully linked CNN includes an additional dense layer. Convolutional layers use kernels to extract features from input, and layers that pool information may enhance these features [245].
ii. *RNNs* are neural networks that excel at processing sequential data, such as time series and natural language [246]. They are made up of several layers, including recurrent layers that remember previous inputs and outputs and fully connected layers that make predictions, as shown in Figure 11. Thus, it is a successful approach in language processing settings that can capture the incoming data's sequential data and then make predictions using an internal storage device that holds a sequence of inputs [247].
iii. *GANs:* These are neural networks that are used to generate new samples that are similar to a given dataset [242]. They are made up of two parts: a generator that produces new

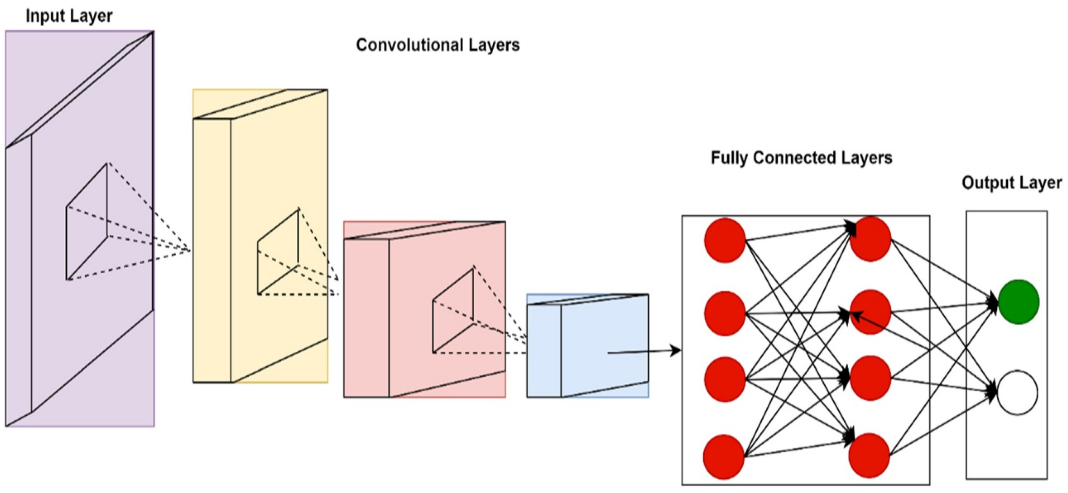

FIGURE 10 Convolutional Neural Network (CNN).

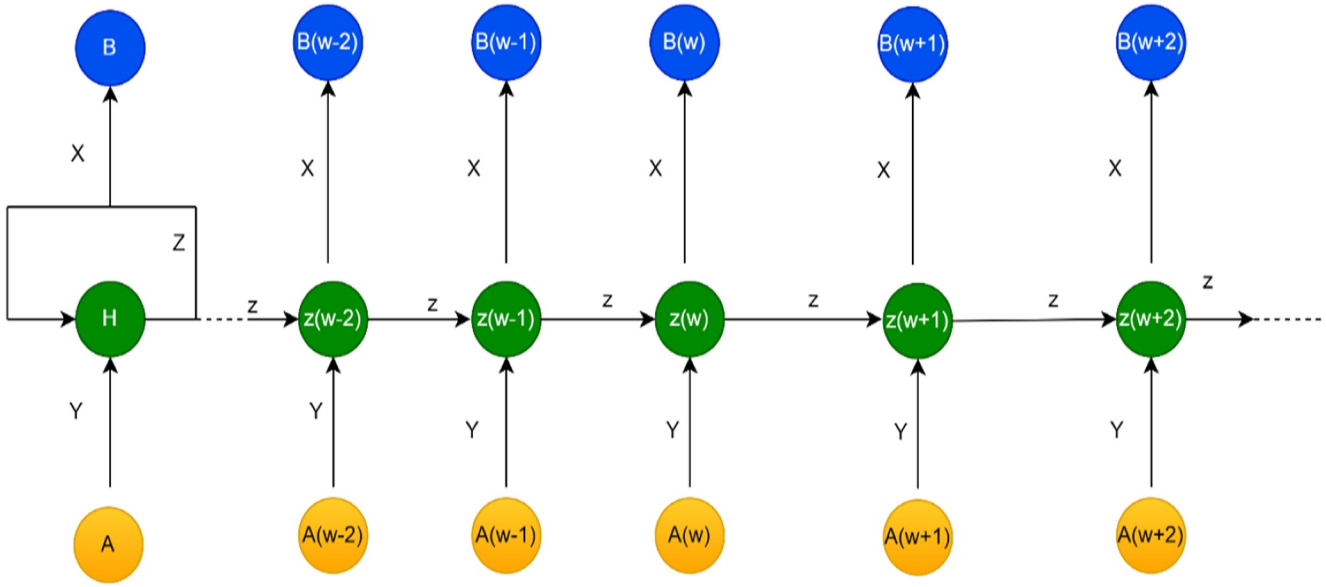

FIGURE 11 Recurrent Neural Networks (RNNs).

samples and a discriminator that attempts to distinguish between real and generated samples. The discriminator model is trained to differentiate between the fake generator's data and the real data sample [248]. In contrast, the generator model generates fake data at the start. The discriminator learns to identify the bogus data, which is then passed on to the generator model to update the model [249], as shown in Figure 6.

iv. *Autoencoders* are neural networks that are used to learn a precise representation of a dataset, known as the bottleneck or latent representation [250]. They are comprised of two parts: an encoder that maps the input to the bottleneck and a decoder that maps the bottleneck back to the input, as shown in Figure 12. It is a feedforward neural network with comparable input and output. They are trained to rebuild the same output data from the input data, that is, to take input data and transform it into a different form [251]. It is a synthetic neural network that has been trained to learn and recreate feature representations and is divided into two sections: an encoder and a decoder, with the former responsible for extracting and the latter responsible for recreating a predefined number of features from the dataset [218]. Whenever the number of nodes in the hidden layer is smaller compared to the number of nodes in the input portion of the layer, the model is capable of compressing data. Thus, the model will learn during training to provide a lower-dimensional representation of the initial input with the smallest amount of information loss [252].

v. *Transformers* are neural networks used mostly for tasks related to natural language processing, such as machine translation, text summarisation, and language modelling [253]. They are composed of several layers, including multi-head self-attention layers that weigh the importance of different words in the input and feed-forward layers that make predictions [254].

7 | BIBLIOMETRIC ANALYSIS OF THE RELATED STUDIES

The goal of the systematic literature review is to find the most effective studies, comprehend the research background, identify the current study area in the domain, and recommend future research paths. This systematic literature review uses the preferred Reporting Items for Systematic Reviews and Meta-Analysis structure method, which includes identifying research studies, filtering research studies, assessing research study eligibility, and including research study analysis [255]. In most situations, the process used for bibliometric studies follows a similar pattern, which includes designing research keywords for the search, obtaining research results, constructing the dataset, describing the findings, and assessing the findings [256]. Though several software tools are used for conducting bibliometric analysis, such as BibExcel, Gephi, Pajek, Excel, HistCite, R-bibliometric package, and VOS-viewer [257], a VOS-viewer was adopted in these studies.

According to the Scopus database analysis, the first search terms selected, "Intrusion Detection System" AND "Cluster-Based WSNs," yielded only 14 documents. The second search keywords used for the Scopus database analysis, "Intrusion Detection System" AND "Wireless Sensor Networks," yielded a total of 841 documents, which were filtered and reduced to a total of 647 documents by applying the inclusion and exclusion criteria shown in Table 3. Many academics are engaged in this study area as a result of the importance of security in countering the daily development of security risks in the application area of WSNs. The analyser effectively picked 10 countries with the greatest research documents as shown in Figure 13: India has the most documents on intrusion detection systems in WSNs, followed by China and the United States, while Pakistan has the fewest, trailed by Italy and Malaysia.

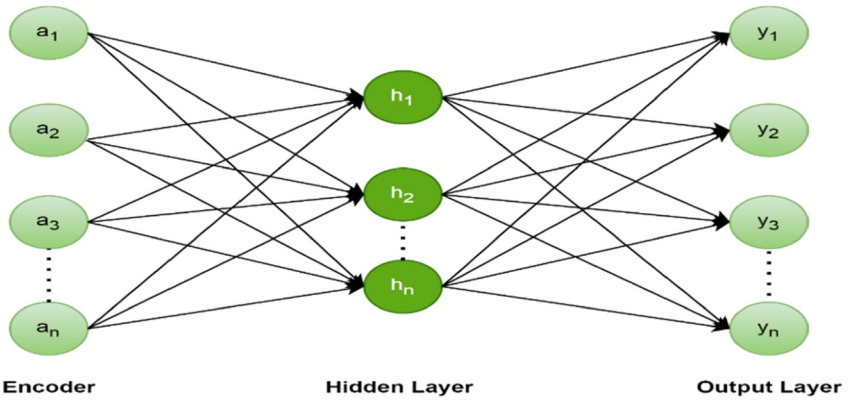

FIGURE 12 The structure of the auto-encoder.

TABLE 3 Search keys for the scopus database analysis.

S/N	Search keys	Documents	Inclusion	Exclusion	Range
1.	"Intrusion detection system" AND "cluster-based wireless sensor networks"	14	All	Nil	All
2.	"Intrusion detection system" AND "wireless sensor networks"	841	✓	✓	✓
		814	Computer science, engineering, Mathematics and decision science	Others	✓
		783	Computer science and engineering	Others	✓
		647	Article, conference paper, conference review, & review	Book chapter, Retracted, book	✓
				All source	Lecture notes
				Final publication stage	Article in press
				All keyword	Nil
				All affiliation	Nil
				All funding sponsor	Nil
				All country	Nil
				Journal & conference proceeding	Book series & trade journal
				English	Chinese, Russian & Turkish

Figure 14 shows that the years 2020 to 2023 have the most publications in the research area, indicating that it is an active research area, and Figure 15 shows that articles account for 47.6% of the documents, conference papers account for 43.3%, conference reviews account for 7.3%, and review accounts for 1.9%. Figure 16 shows that computer science accounts for 42.4% of the documents, followed by engineering, which accounts for 28.2% of the papers. Figure 17 illustrates the author who has the most documents in each country based on document distribution. According to the analysis of documents per year by source, the Journal of Personal Communication is the greatest source of documents in the years 2021 and 2022; from 2023 to the present, IEEE Access is the highest source of documents as shown in Figure 18. By examining the keywords of the articles published and visualising the relationships seen between keywords that are frequently used together, this analysis seeks to comprehend the conceptual framework of the studied research topic.

The VOS-viewer map is generated using bibliographic data derived from the Scopus database. Reading the Scopus bibliographic database files and analysing the data with 647 filter documents using the search keys with inclusion and exclusion as indicated in Table 3. The type of analysis and counting method that was carried out using the VOS viewer is displayed in Table 4.

The co-authorship versus author analysis reveals that for a minimum of three authors and a minimum number of citations per author, there are 562 authors, only six of whom exceed the criterion as shown in Table 5. The total strength

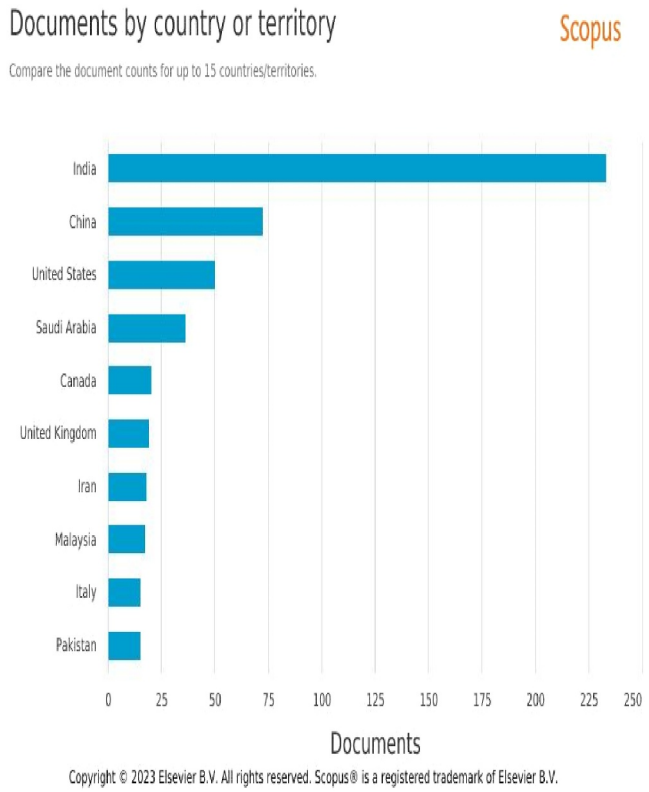

FIGURE 13 Distribution of documents by country.

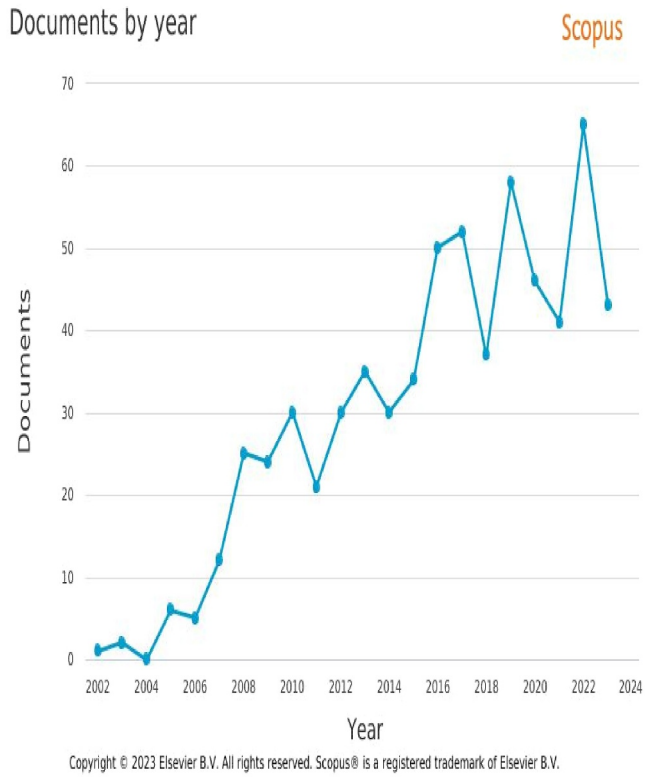

FIGURE 14 Distribution of documents by year.

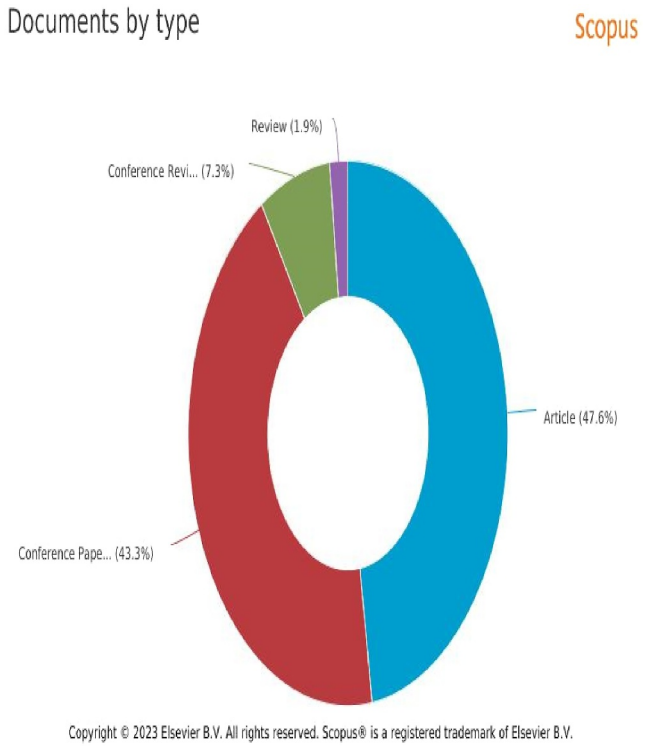

FIGURE 15 Distribution of document by type.

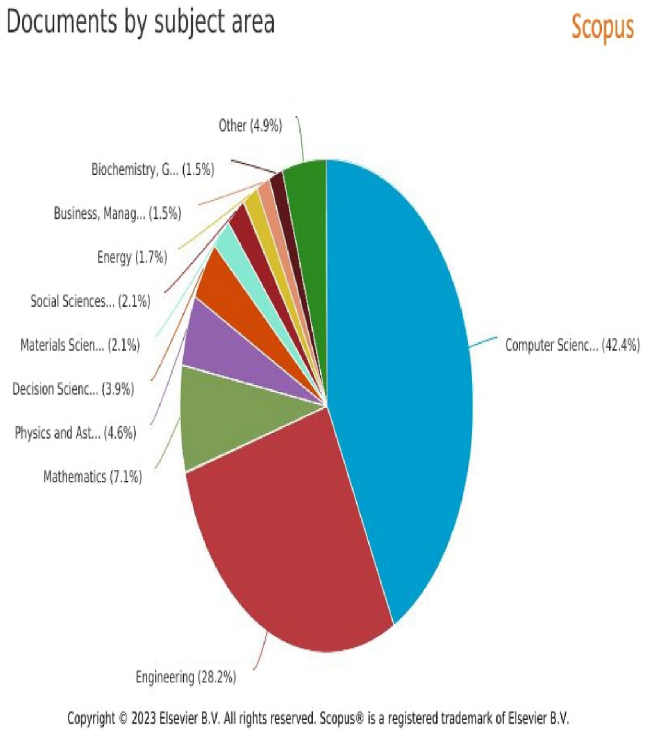

FIGURE 16 Distribution of documents by subject area.

FIGURE 17 Distribution of document by author.

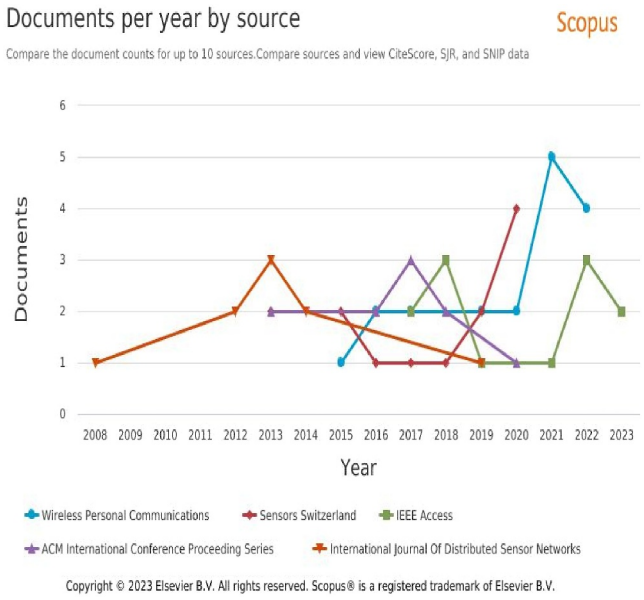

FIGURE 18 Distribution of document by source.

TABLE 4 Type of analysis and counting method.

S/N	Type of analysis	Unit of analysis	Counting method
1.	Co-authorship	Authors	Full counting
2.	✓	Countries	✓
3.	Co-occurrence	All keywords	✓
4.	✓	Author keywords	✓
5.	✓	Index keywords	✓

of co-authorship links with other authors was calculated for each of the six authors, and the authors with the greatest total link strength were chosen by the VOS-viewer analysis, which also shows that there were 28 items with 8 clusters having 76 links and a total link strength of 125. The Co-authorship versus countries research reveals that just 29 nations meet the standards for a minimum of 5 documents from a country with a minimum of 65 citations from a country in 67 countries. The overall strength of co-authorship ties with other countries is determined for each of the 29 countries, and the countries with the largest total link strength are chosen, as seen in the network visualisation illustrated in Figure 19. There are 28 objects, with 76 links in 8 clusters and a total link strength of 125.

The Co-occurrence versus All keywords analysis reveals that a minimum of 5 occurrences of a keyword generates 3655 keywords, 283 of which fulfil the criteria. Thus, the total strength of co-occurrence links with other keywords was calculated for each of the 283 keywords, and the term with the greatest total link strength was chosen. Figure 20 depicts 283 objects, with 11 clusters containing 9463 links and a total link strength of 31,573. According to the Co-occurrence versus Author keyword analysis, a minimum of 5 occurrences of a term yields 1195 keywords, of which 67 fulfil the criterion. Thus, the total strength of co-occurrence links with other keywords was calculated for each of the 67 keywords, and the term with the greatest total link strength was chosen. Figure 21 depicts 67 elements, with 8 clusters with 546 links and a total link strength of 1512. The co-occurrence versus index keyword analysis reveals that a term with a minimum of 5 occurrences yields 3031 keywords, of which 243 fulfil the criterion. Thus, the total strength of the co-occurrence links with other keywords was calculated for each of the 243 keywords, and the term with the greatest total link strength was chosen. As illustrated in Figure 22, there are 243 items, with 9 clusters with 7198 links and a total link strength of 24,598. According to the Citation versus Source analysis, a phrase with a minimum of 5 document sources and a minimum of 2 source citations returned 438 sources, 15 of which met the criterion. As a result, the total strength of the citation ties with other sources was determined, and the sources with the highest total link strength were chosen as shown in Figure 23.

Moreover, from the co-occurrence keywords used, anomaly-based intrusion detection, dimensionality reduction, and network intrusion are some of the domain's researchable key areas because they have fewer occurrences of the author's documents and weaker co-citation strengths. The first search terms utilised in this analysis, "Intrusion Detection System" and "Cluster-Based WSNs," reveal only 14 results in the Scopus database, indicating that the area has yet to be fully explored. There are few authors with high citations in the field, but many authors are currently working in the research area, and we chose the authors with 50 minimum number of citations as the top-cited authors in the field. It reveals that 18,984 authors, out of which 52 meet the threshold, and their network connections show that most of the authors have connections to one another based on their distributions and key research topics depicted in Figure 24.

S/N	Author	Document	Citation	Total link strength
1.	Murugesh.C, Murugan.S	3	3	0
2.	Subramani. S, Selvi. M	4	6	0
3.	Sedjelmaci. H, Senouci. S.M, Feham. M.	3	50	0
4.	Kalnoor. G, Agarkhed. J.	4	54	0
5.	Chen. R.C, Hsieh. C.F, Huang. Y.F.	3	62	0
6.	Otoum. S., Kantarci. B., Mouftah. H	3	134	0

TABLE 5 Co-authorship versus authors threshold analysis.

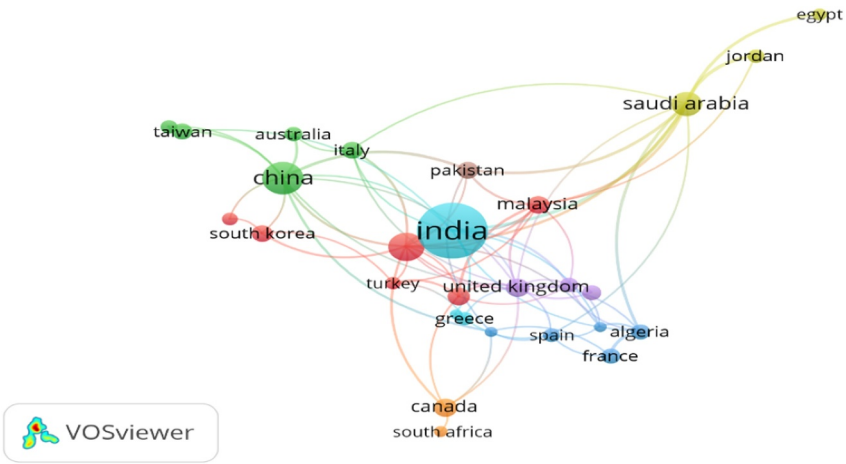

FIGURE 19 Co-authorship versus countries network visualisation.

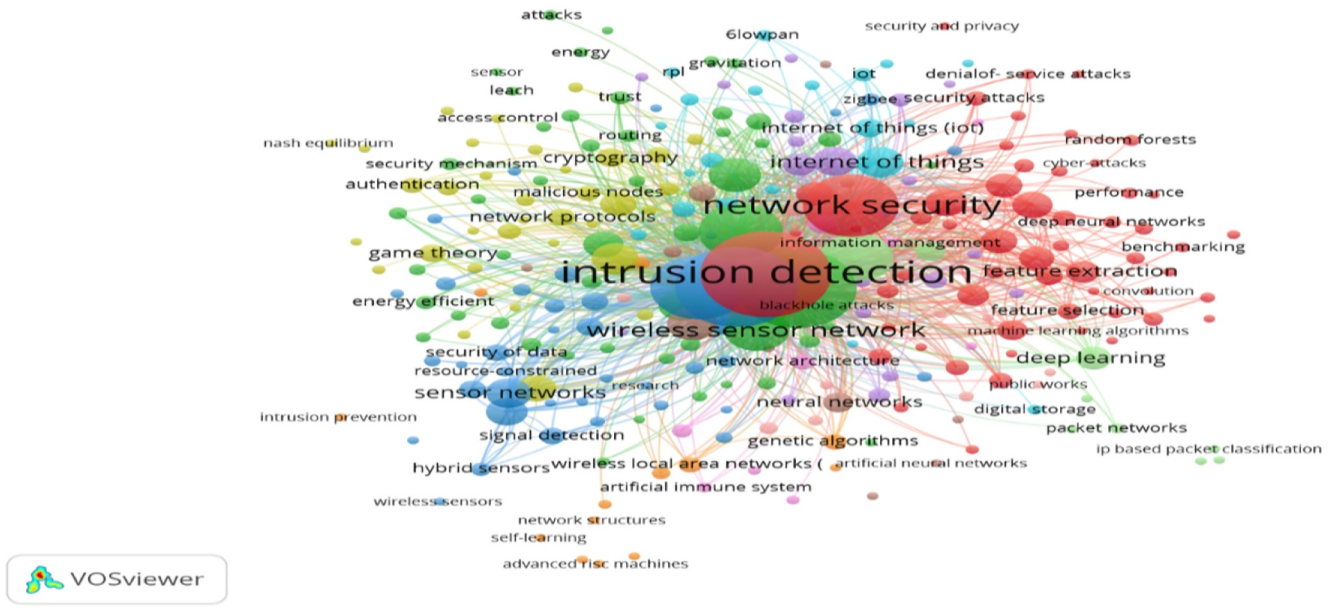

FIGURE 20 Co-occurrence versus all keywords network visualisations.

8 | DISCUSSION ON THE RESEARCH FINDINGS

Various problems have been identified in existing research works on intrusion detection using machine learning algorithms, such as noise, high dimensionality, class imbalance, bias issues with new threat detection, poor feature selection, limited storage, deviation of learning patterns, high computational complexity, sparsity problem, and spatial penalty, among others, which have resulted in high false alarms, low accuracy, low detection, and overfitting of the IDS. According to the existing research, no feature selection process outperformed all classifier algorithms equally well. Most IDS models created contain one or more defects due to issues inherited from the

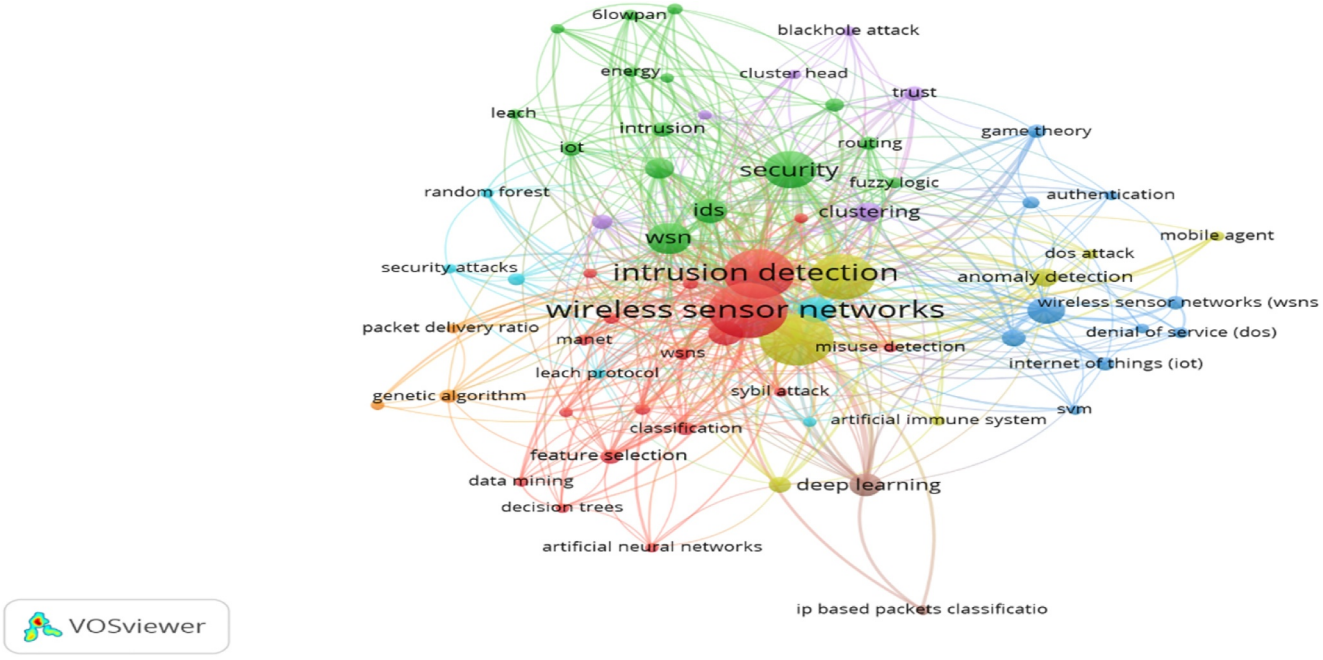

FIGURE 21 Co-occurrence versus author keywords network visualisations.

methodologies utilised, such as feature techniques or classifier algorithms. It was revealed that the feature methodology and feature scaling method utilised significantly impacted the classifier algorithm's performance.

The accuracy of the prediction is governed by the quality of the training dataset and the feature selection technique, which assesses if the chosen feature tends to reflect the whole feature in the original dataset without bias. To overcome some of the shortcomings, some researchers have attempted to improve detection accuracy from support vector machines (SVM) to DT, RF, and DNNs (DNN), as well as from SVM to extreme learning machines, kernel-ELM (K-ELM), and multi-KELM IDS models. While some researchers attempted to optimise from SVM to weighted SVM (WSVM), ABC to improved ABC (IABC), DNN to correlation process with the DNN (CP-DNN), WOA to NBC, and RF to RF-multi-class classifier (RFMC), all of these IDS models were developed or optimised either as signature attack IDS models or anomaly attack IDS models to detect one class of the attacks from other.

Although the cluster-based WSN has various applications in the field of human efforts, new forms of attacks are launched on it regularly to break network security. For example, node energy depletion disconnects nodes from the coverage region, isolating the monitored area. An attacker can launch a DoS attack to cause energy depletion and shorten the network's lifespan.

The study of the examined existing works, as shown in the comparative analysis of Tables 1 and 2, reveals that different researchers used a variety of strategies. Still, none of them outperformed all algorithms equally well. Though an autoencoder outperformed other feature selection techniques regarding dimensionality reduction and class imbalance problems, none of the methods considered the autoencoder's spatiality inefficiency, which harms model accuracy in many cases. A hybrid encoding approach is utilised to circumvent such flaws.

9 | CHALLENGES AND FUTURE RESEARCH DIRECTIONS

The increasing use of WSNs in various fields of human endeavour has resulted in a large number of novel network attacks by cybercriminals. Researchers have worked tirelessly to address security challenges with IDS models. The three techniques used for IDS system design are misuse/signature-based, anomaly-based, and specification-based, but each has significant downsides. The misuse technique cannot detect a novelty attack whose signature pattern is not found in its database and requires the signature pattern in the database to be updated consistently. An intrusion that has no causal effect on the system's normal behaviour cannot be detected by the anomaly technique, and a regular event that may have a causal impact on the system's normal behaviour may be classified as an attack. The specification-based technique is time-consuming and expensive to implement because it must be manually configured to the system specification; as a result of these significant drawbacks, many faults in the IDS model developed by some researchers, particularly low accuracy, high false alarm rate, high computational complexity, high bias classification, and overfitting.

These problems can only be adequately addressed by looking into their root causes. Several researchers are currently working in the field of intrusion detection systems on WSNs, and many research works in this related area have been published. Still, there are only a few publications on intrusion

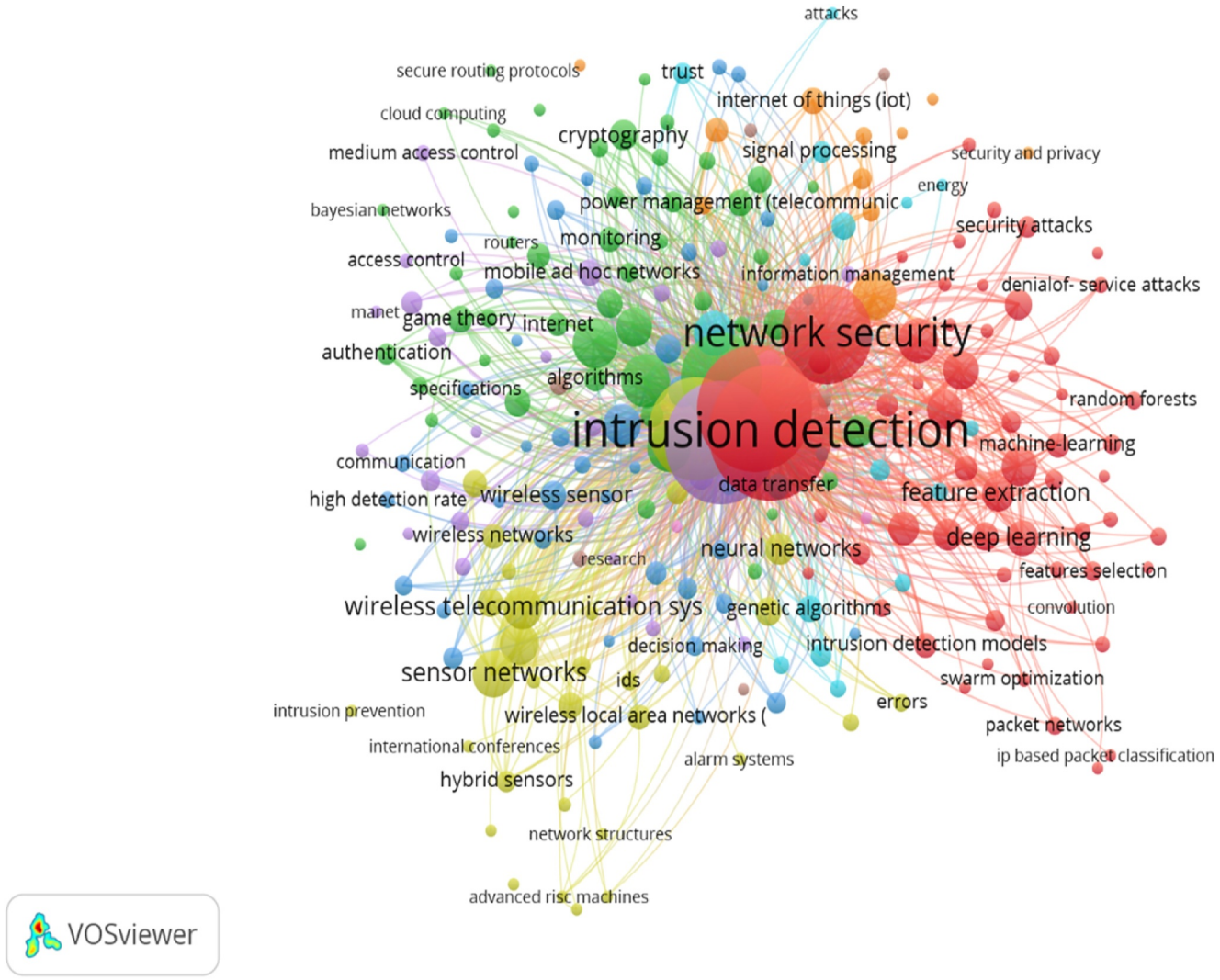

FIGURE 22 Co-occurrence versus index keywords network visualisations.

detection systems on CBWSN, mainly using machine learning approaches for anomaly detection. However, security is a significant concern in WSN applications. The research findings indicate that dimensionality reduction, anomaly detection, and network intrusion detection systems are some of the few keywords in the research areas that have not been fully exploited to address the security as mentioned above challenges, which are indicated as some of the grey researchable regions. As a result, the following are some of the prevalent issues identified in this connected research area.

When nodes in a Cluster-Based WSN (CBWSN) are isolated from their cluster, they can be a root cause of DoS attacks such as grayhole attacks, blackhole attacks, scheduling attacks, flooding attacks, and so on, among the regular network traffic flows in the CBWSN. As a result, a poor detection rate of such attacks may fail network coverage or lower the CBWSN's network lifetime. Researchers have developed an IDS model for detecting DoS classes of attack. However, it has resulted in low detection and high computational complexity with high false alarms.

Another fundamental cause of the underlined problem is the dimensionality reduction or feature extraction problem. Although an autoencoder has promising dimensionality reduction advantages over other techniques used for feature selection in IDS model development, it creates high collinearity between features as a result of the dummy variable trap caused by spatiality inefficiency, which can lead to overfitting of the IDS model in caution; this can affect the accuracy and detections of network traffic attacks, as well as the classifier algorithm's results may be biased due to the problem produced by the complex model. This problem would make detecting several attacks at different locations in the network traffic flow challenging, thereby leading to low detection accuracy [258-268].

The dataset pre-processing phase in the development of the NIDS is essential in achieving a quality feature selection that can improve the performance of the NIDS model, notably its classification algorithm, but several of the methods used have produced some relatively severe flaws such as computational complexity, dimensionality problems, spatial inefficiency and over fittings.

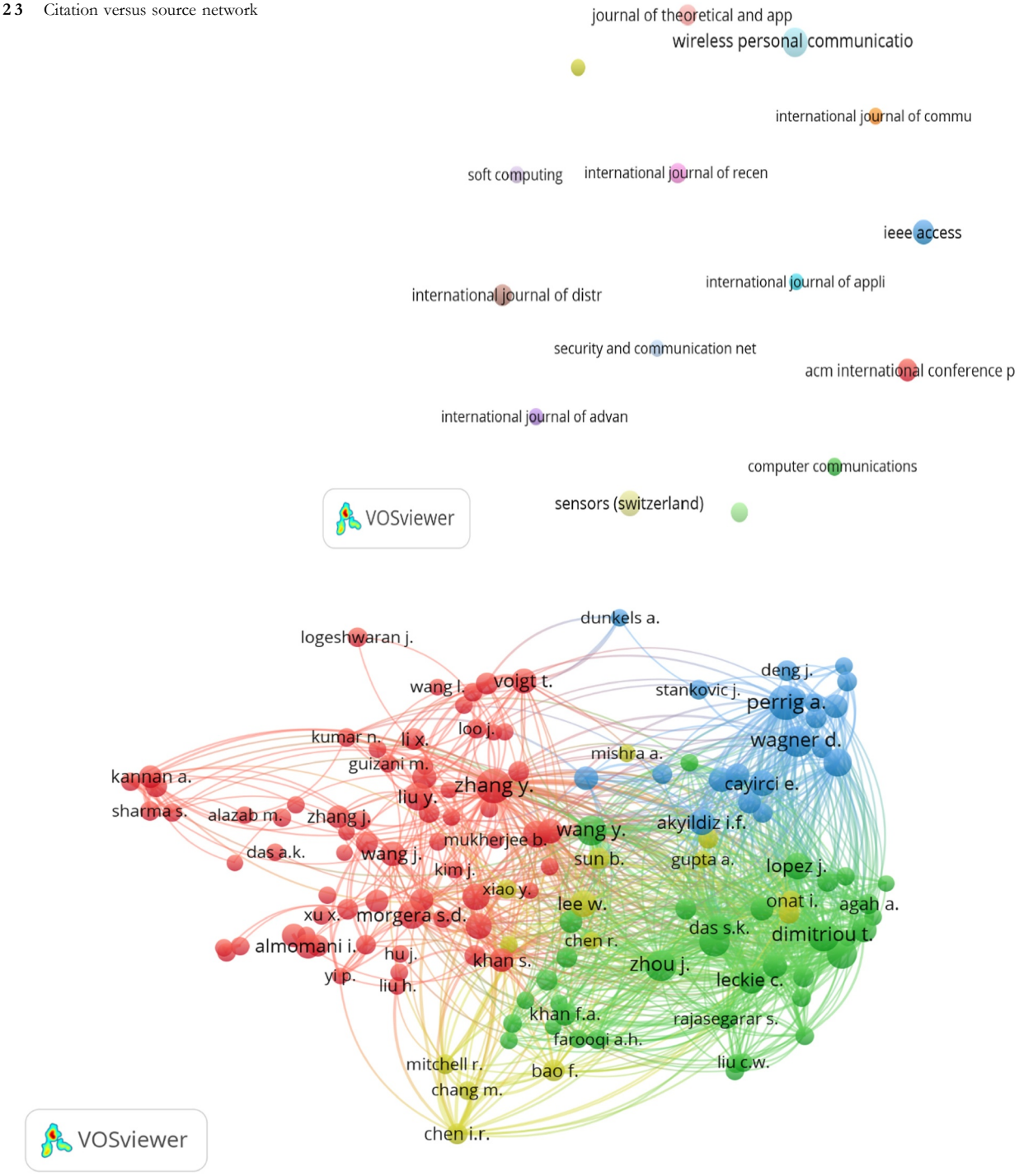

FIGURE 23 Citation versus source network visualisations.

FIGURE 24 Co-citation versus cited authors network visualisations.

10 | CONCLUSION

This paper provided a critical examination of the IDS on CBWSN. It is a system designed to remotely monitor and control a security phenomenon by detecting anomaly attacks in a WSN using an IDS model. The authors depicted an overview of CBWSN in terms of key research areas, optimised clustering algorithm techniques used in the related works, and the major network architectures. We performed a comparative analysis of IDS based on the NSL-KDD dataset as the popular benchmark dataset that constitutes an appropriate proportion of normal network traffic flows with several classes of attack. Thus, we discovered that the feature selection techniques used have a major influence on the efficiency of the classifier algorithm used and that the feature scaling method used has a significant impact on the model's accuracy. The quality of the dataset used in model training can also actually impact the system's general performance. Despite the fact that numerous scientists are attempting

to improve the accuracy of intrusion detection models by developing feature selection techniques that use various algorithms, we recommend the development of approaches that can automate the feature selection process for better IDS model performance. In addition, a bibliometric analysis of related works was performed to supplement the research findings.

We retrieved documents from three different databases: Scopus, Web of Science, and Dimension. We decided to carry out a bibliometric analysis on the Scopus database since it has more resource documents compared with other databases. The bibliometric indicator identifies leading trends in the field of IDS, and we used VOS-viewer to develop the spatial mapping of the documents with the co-authorship, co-occurrence and citation; we generate the network visualisation in terms of the unit of the analysis.

The findings show that anomaly-based intrusion detection, dimensionality reduction, and network intrusion are the domain's researchable key areas, with fewer publications and weaker co-citation strengths. Despite the fact that the research was conducted independently on individual collections from each database, the Scopus collection was considered the main collection because the majority of the collections are also indexed in Scopus.

The purpose of this research paper is to generate relevant findings and a research problem formulation that can lead to a research gap in the research topic's domain area.

AUTHOR CONTRIBUTIONS

Ayuba John: Conceptualisation; methodology; writing - original draft. **Ismail Fauzi Isnin:** Funding acquisition; investigation; resources; supervision; writing - review & editing. **Syed Madni Hamid Hussain:** Funding acquisition; investigation; supervision; writing - review & editing. **Muhammad Faheem:** Methodology; data curation; data validation; writing - review & editing.

ACKNOWLEDGEMENT

The authors appreciate the Nigerian Petroleum Technology Development Fund agency for providing the student with the scholarship to pursue studies in this research field. A heartfelt appreciation goes to the Universiti Teknologi Malaysia and the University of Vaasa Finland for the support to complete this research study.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The datasets will be available upon request to the corresponding author.

ORCID

Ayuba John https://orcid.org/0000-0003-0496-765X
Ismail Fauzi Bin Isnin https://orcid.org/0000-0002-9765-3491
Syed Hamid Hussain Madni https://orcid.org/0000-0002-3816-1382
Muhammed Faheem https://orcid.org/0000-0003-4628-4486

REFERENCE

1. Tirani, SP, Avokh, A, Azar, S: WDAT-OMS: A two-level scheme for efficient data gathering in mobile-sink wireless sensor networks using compressive sensing theory. IET Communications 14(11), 1826–37 (2020). https://doi.org/10.1049/iet-com.2019.0433
2. John A, Igimoh JA. IMPLEMENTATION OF WIRELESS SENSOR NETWORKS FOR REAL TIME MONITORING OF OIL AND GAS FLOW RATE METERING INFRASTRUCTURE. 2017
3. Kocher, P, Horn, J, Fogh, A, Genkin, D, Gruss, D, Haas, W, Hamburg, M., Lipp, M., Mangard, S., Prescher, T., Schwarz, M., Yarom, Y.: Spectre attacks: Exploiting speculative execution. Communications of the ACM 63(7), 93–101 (2020). https://doi.org/10.1145/3399742
4. Zhou, Y, Cheng, G, Jiang, S, Dai, M: Building an efficient intrusion detection system based on feature selection and ensemble classifier. Computer networks 174, 107247 (2020). https://doi.org/10.1016/j.comnet.2020.107247
5. Saranya, T, Sridevi, S, Deisy, C, Chung, TD, Khan, MA: Performance analysis of machine learning algorithms in intrusion detection system: a review. Procedia Computer Science 171, 1251–60 (2020). https://doi.org/10.1016/j.procs.2020.04.133
6. Mutlag, AA, Abd Ghani, MK, Na, Arunkumar, Mohammed, MA, Mohd, O: Enabling technologies for fog computing in healthcare IoT systems. Future Generation Computer Systems 90, 62–78 (2019). https://doi.org/10.1016/j.future.2018.07.049
7. Li, Y, Qin, T, Huang, Y, Lan, J, Liang, Z, Geng, T: HDFEF: A hierarchical and dynamic feature extraction framework for intrusion detection systems. Computers & Security 121, 102842 (2022). https://doi.org/10.1016/j.cose.2022.102842
8. Elsaid, SA, Albatati, NS: An optimized collaborative intrusion detection system for wireless sensor networks. Soft Computing 24(16), 1–15 (2020). https://doi.org/10.1007/s00500-020-04695-0
9. Sarkar, A, Senthil Murugan, T: Cluster head selection for energy efficient and delay-less routing in wireless sensor network. Wireless Networks 25(1), 303–20 (2019). https://doi.org/10.1007/s11276-017-1558-2
10. Amer, R, Butt, MM, Bennis, M, Marchetti, N: Inter-cluster cooperation for wireless D2D caching networks. IEEE Transactions on Wireless Communications 17(9), 6108–21 (2018). https://doi.org/10.1109/twc.2018.2854603
11. El Khediri, S, Fakhet, W, Moulahi, T, Khan, R, Thaljaoui, A, Kachouri, A: Improved node localization using K-means clustering for Wireless Sensor Networks. Computer Science Review 37, 100284 (2020). https://doi.org/10.1016/j.cosrev.2020.100284
12. Prabha, M, Darly, SS, Rabi, BJ: A novel approach of hierarchical compressive sensing in wireless sensor network using block tri-diagonal matrix clustering. Computer Communications 168, 54–64 (2021). https://doi.org/10.1016/j.comcom.2020.12.017
13. Kowsalya, R, Jeetha, BR: Cluster based data-aggregation using light-weight cryptographic algortihm for wireless sensor networks. Materials Today: Proceedings (2021)
14. Sert, SA, Yazici, A: Increasing energy efficiency of rule-based fuzzy clustering algorithms using CLONALG-M for wireless sensor networks. Applied Soft Computing 109, 107510 (2021). https://doi.org/10.1016/j.asoc.2021.107510
15. Sharma, R, Vashisht, V, Singh, U: Fuzzy modelling based energy aware clustering in wireless sensor networks using modified invasive weed optimization. Journal of King Saud University-Computer and Information Sciences 34(5), 1884–1894 (2019). https://doi.org/10.1016/j.jksuci.2019.11.014
16. Fanian, F, Rafsanjani, MK: A new fuzzy multi-hop clustering protocol with automatic rule tuning for wireless sensor networks. Applied Soft Computing 89, 106115 (2020). https://doi.org/10.1016/j.asoc.2020.106115
17. Esmaeili, H, Bidgoli, BM, Hakami, V: CMML: Combined meta-heuristic-machine learning for adaptable routing in clustered wireless

sensor networks. Applied Soft Computing 118, 108477 (2022). https://doi.org/10.1016/j.asoc.2022.108477
18. Liu, X, Yu, J, Zhang, W, Tian, H: Low-energy dynamic clustering scheme for multi-layer wireless sensor networks. Computers & Electrical Engineering 91, 107093 (2021). https://doi.org/10.1016/j.compeleceng.2021.107093
19. Bohra, B, Kumar, S, Jain, A, Aggarwal, S, Gupta, MK: Achieving uneven clustering in wireless sensor networks using fuzzy logic. Materials Today: Proceedings 51, 2495–2499 (2021). https://doi.org/10.1016/j.matpr.2021.11.629
20. Dawood, MS, Benazer, SS, Saravanan, SV, Karthik, V: Energy efficient distance based clustering protocol for heterogeneous wireless sensor networks. Materials Today: Proceedings 45, 2599–602 (2021). https://doi.org/10.1016/j.matpr.2020.11.339
21. Kaur, S, Mir, RN, Khamparia, A, Rani, P, Gupta, D, Khanna, A: Heterogeneous load balancing clustering protocol for Wireless Sensor Networks. Cognitive Systems Research 70, 10–7 (2021). https://doi.org/10.1016/j.cogsys.2021.07.001
22. Yalçın, S, Erdem, E: TEO-MCRP: Thermal exchange optimization-based clustering routing protocol with a mobile sink for wireless sensor networks. Journal of King Saud University-Computer and Information Sciences 34(8), 5333–5348 (2022). https://doi.org/10.1016/j.jksuci.2022.01.007
23. Gbadouissa, JEZ, Ari, AAA, Titouna, C, Gueroui, AM, Thiare, O: HGC: HyperGraph based Clustering scheme for power aware wireless sensor networks. Future Generation Computer Systems 105, 175–83 (2020). https://doi.org/10.1016/j.future.2019.11.043
24. Ramani, G, Amarendra, K: Optimal path selection with clustering in wireless sensor networks. Materials Today: Proceedings (2021)
25. Gopinath, S, Kumar, KV, Elayaraja, P, Parameswari, A, Balakrishnan, S, Thiruppathi, M: Sceer: secure cluster based efficient energy routing scheme for wireless sensor networks. Materials Today: Proceedings 45, 3579–84 (2021). https://doi.org/10.1016/j.matpr.2020.12.1096
26. Saidi, A, Benahmed, K, Seddiki, N: Secure cluster head election algorithm and misbehavior detection approach based on trust management technique for clustered wireless sensor networks. Ad Hoc Networks 106, 102215 (2020). https://doi.org/10.1016/j.adhoc.2020.102215
27. Deepa, O, Suguna, J: An optimized QoS-based clustering with multi-path routing protocol for wireless sensor networks. Journal of King Saud University-Computer and Information Sciences 32(7), 763–74 (2020). https://doi.org/10.1016/j.jksuci.2017.11.007
28. Prakash, PS, Kavitha, D, Reddy, PC: Delay-aware relay node selection for cluster-based wireless sensor networks. Measurement: Sensors 24, 100403 (2022). https://doi.org/10.1016/j.measen.2022.100403
29. Al-Sulaifanie, AI, Al-Sulaifanie, BK, Biswas, S: Recent trends in clustering algorithms for wireless sensor networks: A comprehensive review. Computer Communications 191, 395–424 (2022). https://doi.org/10.1016/j.comcom.2022.05.006
30. Amutha, J, Sharma, S, Sharma, SK: Strategies based on various aspects of clustering in wireless sensor networks using classical, optimization and machine learning techniques: Review, taxonomy, research findings, challenges and future directions. Computer Science Review 40, 100376 (2021). https://doi.org/10.1016/j.cosrev.2021.100376
31. Kumar, N, Kumar, H: A fuzzy clustering technique for enhancing the convergence performance by using improved Fuzzy c-means and Particle Swarm Optimization algorithms. Data & Knowledge Engineering 140, 102050 (2022). https://doi.org/10.1016/j.datak.2022.102050
32. Wohwe Sambo, D, Yenke, BO, Förster, A, Dayang, P: Optimized clustering algorithms for large wireless sensor networks: A review. Sensors 19(2), 322 (2019). https://doi.org/10.3390/s19020322
33. Yusuf, MN, Bakar, KbA, Isyaku, B, Osman, AH, Nasser, M, Elhaj, FA: Adaptive Path Selection Algorithm with Flow Classification for Software-Defined Networks. Mathematics 11(6), 1404 (2023). https://doi.org/10.3390/math11061404
34. Rawat, P, Chauhan, S: Clustering protocols in wireless sensor network: A survey, classification, issues, and future directions. Computer Science Review 40, 100396 (2021). https://doi.org/10.1016/j.cosrev.2021.100396
35. Sharma, D, Ojha, A, Bhondekar, AP: Heterogeneity consideration in wireless sensor networks routing algorithms: a review. The journal of supercomputing 75(5), 2341–94 (2019). https://doi.org/10.1007/s11227-018-2635-8
36. Elsmany, EFA, Omar, MA, Wan, T-C, Altahir, AA: EESRA: Energy efficient scalable routing algorithm for wireless sensor networks. IEEE Access 7, 96974–83 (2019). https://doi.org/10.1109/access.2019.2929578
37. Zakariayi, S, Babaie, S: DEHCIC: A distributed energy-aware hexagon based clustering algorithm to improve coverage in wireless sensor networks. Peer-to-Peer Networking and Applications 12(4), 689–704 (2019). https://doi.org/10.1007/s12083-018-0666-9
38. Arjunan, S, Pothula, S: A survey on unequal clustering protocols in wireless sensor networks. Journal of King Saud University-Computer and Information Sciences 31(3), 304–17 (2019). https://doi.org/10.1016/j.jksuci.2017.03.006
39. Khan, FA, Khan, M, Asif, M, Khalid, A, Haq, IU: Hybrid and multi-hop advanced zonal-stable election protocol for wireless sensor networks. IEEE Access 7, 25334–46 (2019). https://doi.org/10.1109/access.2019.2899752
40. Maheshwari, P, Sharma, AK, Verma, K: Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization. Ad Hoc Networks 110, 102317 (2021). https://doi.org/10.1016/j.adhoc.2020.102317
41. Madni, SHH, Abd Latiff, MS, Coulibaly, Y, Abdulhamid, S.M.: Resource scheduling for infrastructure as a service (IaaS) in cloud computing: Challenges and opportunities. Journal of Network and Computer Applications 68, 173–200 (2016). https://doi.org/10.1016/j.jnca.2016.04.016
42. Moussa N, Hamidi-Alaoui Z, Alaoui AEBE. EHRP: An effective hybrid routing protocol to compromise between energy consumption and delay in WSNs. arXiv preprint arXiv:220103910. 2022
43. Guleria, K, Verma, AK: Comprehensive review for energy efficient hierarchical routing protocols on wireless sensor networks. Wireless Networks 25(3), 1159–83 (2019). https://doi.org/10.1007/s11276-018-1696-1
44. Madni, SHH, Latiff, MSA, Coulibaly, Y, Abdulhamid, SiM: Recent advancements in resource allocation techniques for cloud computing environment: a systematic review. Cluster Computing 20(3), 2489–533 (2017). https://doi.org/10.1007/s10586-016-0684-4
45. Sajwan, M, Gosain, D, Sharma, AK: CAMP: cluster aided multi-path routing protocol for wireless sensor networks. Wireless Networks 25(5), 2603–20 (2019). https://doi.org/10.1007/s11276-018-1689-0
46. John, A, Isnin, IF, Madni, SHH: Current Security Threats in Applications of Wireless Sensor Network. International Journal on Engineering, Science and Technology 5(3), 255–72 (2023). https://doi.org/10.46328/ijonest.174
47. Ayuba, J, Safwana, H, Abdulazeez, Y, Ma'azu, D: Software Development of Integrated Wireless Sensor Networks for RealTime Monitoring of Oil and Gas Flow Rate Metering Infrastructure. Journal of Information Technology & Software Engineering 8(2), 1–10 (2018)
48. Jiang, J, Wang, H, Mu, X, Guan, S: Logistics industry monitoring system based on wireless sensor network platform. Computer Communications 155, 58–65 (2020). https://doi.org/10.1016/j.comcom.2020.03.016
49. Boubiche, S, Boubiche, DE, Bilami, A, Toral-Cruz, H: Big data challenges and data aggregation strategies in wireless sensor networks. IEEE access 6, 20558–71 (2018). https://doi.org/10.1109/access.2018.2821445
50. Farahzadi, HR, Langarizadeh, M, Mirhosseini, M, Fatemi Aghda, SA: An improved cluster formation process in wireless sensor network to decrease energy consumption. Wireless Networks 27(2), 1077–87 (2021). https://doi.org/10.1007/s11276-020-02485-y
51. Shahraki, A, Taherkordi, A, Haugen, Ø, Eliassen, F: Clustering objectives in wireless sensor networks: A survey and research direction analysis. Computer Networks 180, 107376 (2020). https://doi.org/10.1016/j.comnet.2020.107376
52. Sharma, A, Kumar, R, Kaur, P (eds.): Study of issues and challenges of different routing protocols in wireless sensor network 2019 Fifth

international Conference on image information processing (ICIIP). IEEE (2019)
53. Muthanna, MSA, Muthanna, A, Rafiq, A, Hammoudeh, M, Alkanhel, R, Lynch, S, Abd El-Latif, A.A.: Deep reinforcement learning based transmission policy enforcement and multi-hop routing in QoS aware LoRa IoT networks. Computer Communications 183, 33–50 (2022). https://doi.org/10.1016/j.comcom.2021.11.010
54. Gebremariam, GG, Panda, J, Indu, S: Design of advanced intrusion detection systems based on hybrid machine learning techniques in hierarchically wireless sensor networks. Connection Science 35(1), 2246703 (2023). https://doi.org/10.1080/09540091.2023.2246703
55. Miranda, C, Kaddoum, G, Bou-Harb, E, Garg, S, Kaur, K: A collaborative security framework for software-defined wireless sensor networks. IEEE Transactions on Information Forensics and Security 15, 2602–15 (2020). https://doi.org/10.1109/tifs.2020.2973875
56. Yadav, R, Sreedevi, I, Gupta, D: Augmentation in performance and security of WSNs for IoT applications using feature selection and classification techniques. Alexandria Engineering Journal 65, 461–73 (2023). https://doi.org/10.1016/j.aej.2022.10.033
57. Ahmad, Z, Shahid Khan, A, Wai Shiang, C, Abdullah, J, Ahmad, F: Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies 32(1), e4150 (2021). https://doi.org/10.1002/ett.4150
58. Zhou, L, Guo, H (eds.): Anomaly detection methods for IIoT networks 2018 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI). IEEE (2018)
59. Khonde, S, Ulagamuthalvi, V: Ensemble-based semi-supervised learning approach for a distributed intrusion detection system. Journal of Cyber Security Technology 3(3), 163–88 (2019). https://doi.org/10.1080/23742917.2019.1623475
60. Mourad, A, Tout, H, Wahab, OA, Otrok, H, Dbouk, T: Ad hoc vehicular fog enabling cooperative low-latency intrusion detection. IEEE Internet of Things Journal 8(2), 829–43 (2020). https://doi.org/10.1109/jiot.2020.3008488
61. Sharma, DK, Dhankhar, T, Agrawal, G, Singh, SK, Gupta, D, Nebhen, J, Razzak, I.: Anomaly detection framework to prevent DDoS attack in fog empowered IoT networks. Ad Hoc Networks 121, 102603 (2021). https://doi.org/10.1016/j.adhoc.2021.102603
62. Lawal, MA, Shaikh, RA, Hassan, SR: An anomaly mitigation framework for iot using fog computing. Electronics 9(10), 1565 (2020). https://doi.org/10.3390/electronics9101565
63. Liang, W, Li, K-C, Long, J, Kui, X, Zomaya, AY: An industrial network intrusion detection algorithm based on multifeature data clustering optimization model. IEEE Transactions on Industrial Informatics 16(3), 2063–71 (2019). https://doi.org/10.1109/tii.2019.2946791
64. Dwivedi, S, Vardhan, M, Tripathi, S: Building an efficient intrusion detection system using grasshopper optimization algorithm for anomaly detection, pp. 1–20. Cluster Computing (2021)
65. El Ghazi, A, Rachid, AM (eds.): Machine learning and datamining methods for hybrid IoT intrusion detection 2020 5th international conference on cloud computing and artificial intelligence: technologies and applications (CloudTech). IEEE (2020)
66. Aljawarneh, SA, Vangipuram, R: GARUDA: Gaussian dissimilarity measure for feature representation and anomaly detection in Internet of things. The Journal of Supercomputing 76(6), 4376–413 (2020). https://doi.org/10.1007/s11227-018-2397-3
67. Kolandaisamy, R, Noor, RM, Kolandaisamy, I, Ahmedy, I, Kiah, MLM, Tamil, MEM, Nandy, T.: A stream position performance analysis model based on DDoS attack detection for cluster-based routing in VANET. Journal of Ambient Intelligence and Humanized Computing 12(6), 6599–612 (2021). https://doi.org/10.1007/s12652-020-02279-2
68. de Souza, CA, Westphall, CB, Machado, RB, Loffi, L, Westphall, CM, Geronimo, GA: Intrusion detection and prevention in fog based IoT environments: A systematic literature review. Computer Networks 214, 109154 (2022). https://doi.org/10.1016/j.comnet.2022.109154
69. Mehmood, A, Khanan, A, Umar, MM, Abdullah, S, Ariffin, KAZ, Song, H: Secure knowledge and cluster-based intrusion detection mechanism for smart wireless sensor networks. IEEE Access 6, 5688–94 (2017). https://doi.org/10.1109/access.2017.2770020
70. Heidari, A, Jabraeil Jamali, MA: Internet of Things intrusion detection systems: A comprehensive review and future directions, pp. 1–28. Cluster Computing (2022)
71. Herrera-Semenets, V, Bustio-Martínez, L, Hernández-León, R, van den Berg, J: A multi-measure feature selection algorithm for efficacious intrusion detection. Knowledge-Based Systems 227, 107264 (2021). https://doi.org/10.1016/j.knosys.2021.107264
72. Dubey, GP, Bhujade, RK: Optimal feature selection for machine learning based intrusion detection system by exploiting attribute dependence. Materials Today: Proceedings 47, 6325–31 (2021). https://doi.org/10.1016/j.matpr.2021.04.643
73. Alazzam, H, Sharieh, A, Sabri, KE: A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer. Expert systems with applications 148, 113249 (2020). https://doi.org/10.1016/j.eswa.2020.113249
74. Kanna, PR, Santhi, P: Unified deep learning approach for efficient intrusion detection system using integrated spatial–temporal features. Knowledge-Based Systems 226, 107132 (2021). https://doi.org/10.1016/j.knosys.2021.107132
75. Nimbalkar, P, Kshirsagar, D: Feature selection for intrusion detection system in Internet-of-Things (IoT). ICT Express 7(2), 177–81 (2021). https://doi.org/10.1016/j.icte.2021.04.012
76. Selvakumar, K, Karuppiah, M, SaiRamesh, L, Islam, SH, Hassan, MM, Fortino, G, Choo, K.K.R.: Intelligent temporal classification and fuzzy rough set-based feature selection algorithm for intrusion detection system in WSNs. Information Sciences 497, 77–90 (2019). https://doi.org/10.1016/j.ins.2019.05.040
77. Halim, Z, Yousaf, MN, Waqas, M, Sulaiman, M, Abbas, G, Hussain, M, Ahmad, I., Hanif, M.: An effective genetic algorithm-based feature selection method for intrusion detection systems. Computers & Security 110, 102448 (2021). https://doi.org/10.1016/j.cose.2021.102448
78. Li, X, Chen, W, Zhang, Q, Wu, L: Building auto-encoder intrusion detection system based on random forest feature selection. Computers & Security 95, 101851 (2020). https://doi.org/10.1016/j.cose.2020.101851
79. Almasoudy, FH, Al-Yaseen, WL, Idrees, AK: Differential evolution wrapper feature selection for intrusion detection system. Procedia Computer Science 167, 1230–9 (2020). https://doi.org/10.1016/j.procs.2020.03.438
80. Bagui, S, Li, K: Resampling imbalanced data for network intrusion detection datasets. Journal of Big Data 8(1), 1–41 (2021). https://doi.org/10.1186/s40537-020-00390-x
81. Cui, J, Zong, L, Xie, J, Tang, M: A novel multi-module integrated intrusion detection system for high-dimensional imbalanced data. Applied Intelligence 53, 1–17 (2022). https://doi.org/10.1007/s10489-022-03361-2
82. Ahsan, R, Shi, W, Corriveau, JP: Network intrusion detection using machine learning approaches: Addressing data imbalance. IET Cyber-Physical Systems: Theory & Applications 7(1), 30–9 (2022). https://doi.org/10.1049/cps2.12013
83. Soon, HF, Amir, A, Azemi, SN (eds.): An Analysis of Multiclass Imbalanced Data Problem in Machine Learning for Network Attack Detections Journal of Physics: Conference Series. IOP Publishing (2021)
84. Imrana, Y, Xiang, Y, Ali, L, Abdul-Rauf, Z, Hu, Y-C, Kadry, S, Lim, S.: χ 2-BidLSTM: A Feature Driven Intrusion Detection System Based on χ 2 Statistical Model and Bidirectional LSTM. Sensors 22(5), 2018 (2022). https://doi.org/10.3390/s22052018
85. Gowdhaman, V, Dhanapal, R: An intrusion detection system for wireless sensor networks using deep neural network. Soft Computing 26(23), 1–9 (2021). https://doi.org/10.1007/s00500-021-06473-y
86. Amaran, S, Mohan, RM (eds.): Intrusion Detection System using Optimal Support Vector Machine for Wireless Sensor Networks 2021

International Conference on Artificial Intelligence and Smart Systems (ICAIS). IEEE (2021)
87. Alaparthy, VT, Morgera, SD: A multi-level intrusion detection system for wireless sensor networks based on immune theory. IEEE Access 6, 47364–73 (2018). https://doi.org/10.1109/access.2018.2866962
88. Umarani, C, Kannan, S: Intrusion detection system using hybrid tissue growing algorithm for wireless sensor network. Peer-to-Peer Networking and Applications 13(3), 752–61 (2020). https://doi.org/10.1007/s12083-019-00781-9
89. Du, Y, Xia, J, Ma, J, Zhang, W: An Optimal Decision Method for Intrusion Detection System in Wireless Sensor Networks With Enhanced Cooperation Mechanism. IEEE Access 9, 69498–512 (2021). https://doi.org/10.1109/access.2021.3065571
90. Narasimha Prasad, S, Senthamil Selvan, K, Lakshmi Dhevi, B: Intrusion Detection System in Wireless Sensor Networks and Fair Resource Allocation Using Geometric Deep Learning Techniques. Wireless Personal Communications 123(4), 1–12 (2021). https://doi.org/10.1007/s11277-021-09294-2
91. Alruhaily NM, Ibrahim DM. A Multi-layer Machine Learning-based Intrusion Detection System for Wireless Sensor Networks
92. Maheswari, M, Karthika, R: A Novel QoS Based Secure Unequal Clustering Protocol with Intrusion Detection System in Wireless Sensor Networks. Wireless Personal Communications 118(2), 1535–57 (2021). https://doi.org/10.1007/s11277-021-08101-2
93. Kurniawan, MT, Yazid, S (eds.): Mitigation and Detection Strategy of DoS Attack on Wireless Sensor Network Using Blocking Approach and Intrusion Detection System 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE). IEEE (2020)
94. Wang, J, Jiang, S, Fapojuwo, AO: A protocol layer trust-based intrusion detection scheme for wireless sensor networks. Sensors 17(6), 1227 (2017). https://doi.org/10.3390/s17061227
95. Ghugar, U, Pradhan, J, Bhoi, SK, Sahoo, RR: LB-IDS: Securing wireless sensor network using protocol layer trust-based intrusion detection system. Journal of Computer Networks and Communications 2019, 1–13 (2019). https://doi.org/10.1155/2019/2054298
96. Pandey, SK: An anomaly detection technique-based intrusion detection system for wireless sensor network. International Journal of Wireless and Mobile Computing 17(4), 323–33 (2019). https://doi.org/10.1504/ijwmc.2019.10024335
97. Martinez, CV, Vogel-Heuser, B: A host intrusion detection system architecture for embedded industrial devices. Journal of The Franklin Institute 358(1), 210–36 (2021). https://doi.org/10.1016/j.jfranklin.2019.03.037
98. Godala, S, Vaddella, RPV: A study on intrusion detection system in wireless sensor networks. International Journal of Communication Networks and Information Security 12(1), 127–41 (2020). https://doi.org/10.17762/ijcnis.v12i1.4429
99. Riyad, A, Ahmed, MI, Khan, RR: An adaptive distributed intrusion detection system architecture using multi agents. International Journal of Electrical and Computer Engineering 9(6), 4951 (2019). https://doi.org/10.11591/ijece.v9i6.pp4951-4960
100. Almomani, I, Alromi, A: Integrating software engineering processes in the development of efficient intrusion detection systems in wireless sensor networks. Sensors 20(5), 1375 (2020). https://doi.org/10.3390/s20051375
101. Khan, ZA, Herrmann, P: Recent advancements in intrusion detection systems for the Internet of Things. Security and Communication Networks 2019, 1–19 (2019). https://doi.org/10.1155/2019/4301409
102. Colom, JF, Gil, D, Mora, H, Volckaert, B, Jimeno, AM: Scheduling framework for distributed intrusion detection systems over heterogeneous network architectures. Journal of Network and Computer Applications 108, 76–86 (2018). https://doi.org/10.1016/j.jnca.2018.02.004
103. Nguyen, TG, Phan, TV, Nguyen, BT, So-In, C, Baig, ZA, Sanguanpong, S: Search: A collaborative and intelligent nids architecture for sdn-based cloud iot networks. IEEE access 7, 107678–94 (2019). https://doi.org/10.1109/access.2019.2932438
104. Borkar, GM, Patil, LH, Dalgade, D, Hutke, A: A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN: A data mining concept. Sustainable Computing: Informatics and Systems 23, 120–35 (2019). https://doi.org/10.1016/j.suscom.2019.06.002
105. Al, S, Dener, M: STL-HDL: A new hybrid network intrusion detection system for imbalanced dataset on big data environment. Computers & Security 110, 102435 (2021). https://doi.org/10.1016/j.cose.2021.102435
106. Keerthika, M, Shanmugapriya, D: Wireless Sensor Networks: Active and Passive attacks-Vulnerabilities and Countermeasures. Global Transitions Proceedings 2(2), 362–7 (2021). https://doi.org/10.1016/j.gltp.2021.08.045
107. Gavel, S, Raghuvanshi, AS, Tiwari, S: A novel density estimation based intrusion detection technique with Pearson's divergence for Wireless Sensor Networks. ISA transactions 111, 180–91 (2021). https://doi.org/10.1016/j.isatra.2020.11.016
108. Mahdavi, E, Fanian, A, Amini, F: A real-time alert correlation method based on code-books for intrusion detection systems. Computers & Security 89, 101661 (2020). https://doi.org/10.1016/j.cose.2019.101661
109. Singh, A, Nagar, J, Sharma, S, Kotiyal, V: A Gaussian process regression approach to predict the k-barrier coverage probability for intrusion detection in wireless sensor networks. Expert Systems With Applications 172, 114603 (2021). https://doi.org/10.1016/j.eswa.2021.114603
110. Li, Z, Huang, C, Qiu, W: An intrusion detection method combining variational auto-encoder and generative adversarial networks. Computer Networks 253, 110724 (2024). https://doi.org/10.1016/j.comnet.2024.110724
111. Kolukisa, B, Dedeturk, BK, Hacilar, H, Gungor, VC: An efficient network intrusion detection approach based on logistic regression model and parallel artificial bee colony algorithm. Computer Standards & Interfaces 89, 103808 (2024). https://doi.org/10.1016/j.csi.2023.103808
112. Alrayes, FS, Zakariah, M, Amin, SU, Khan, ZI, Alqurni, JS: CNN Channel Attention Intrusion Detection System Using NSL-KDD Dataset. Computers, Materials & Continua 79(3), 4319–4347 (2024). https://doi.org/10.32604/cmc.2024.050586
113. Vibhute, AD, Patil, CH, Mane, AV, Kale, KV: Towards detection of network anomalies using machine learning algorithms on the NSL-KDD benchmark datasets. Procedia Computer Science 233, 960–9 (2024). https://doi.org/10.1016/j.procs.2024.03.285
114. Chelloug, SA: A Robust Approach for Multi Classification-Based Intrusion Detection through Stacking Deep Learning Models. Computers, Materials & Continua 79(3), 4845–4861 (2024). https://doi.org/10.32604/cmc.2024.051539
115. Kanna, PR, Santhi, P: Hybrid Intrusion Detection using MapReduce based Black Widow Optimized Convolutional Long Short-Term Memory Neural Networks. Expert Systems with Applications 194, 116545 (2022). https://doi.org/10.1016/j.eswa.2022.116545
116. Albahar, MA, Binsawad, M, Almalki, J, El-etriby, S, Karali, S: Improving Intrusion Detection System using Artificial Neural Network. International Journal of Advanced Computer Science and Applications 11(6) (2020). https://doi.org/10.14569/ijacsa.2020.0110670
117. Benmessahel, I, Xie, K, Chellal, M: A new evolutionary neural networks based on intrusion detection systems using multiverse optimization. Applied Intelligence 48(8), 2315–27 (2018). https://doi.org/10.1007/s10489-017-1085-y
118. Zhang, Y, Li, P, Wang, X: Intrusion detection for IoT based on improved genetic algorithm and deep belief network. IEEE Access 7, 31711–22 (2019). https://doi.org/10.1109/access.2019.2903723
119. Hajimirzaei, B, Navimipour, NJ: Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm. Ict Express 5(1), 56–9 (2019). https://doi.org/10.1016/j.icte.2018.01.014
120. Safaldin, M, Otair, M, Abualigah, L: Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. Journal of ambient intelligence and humanized computing 12(2), 1559–76 (2021). https://doi.org/10.1007/s12652-020-02228-z

121. Benmessahel, I, Xie, K, Chellal, M, Semong, T: A new evolutionary neural networks based on intrusion detection systems using locust swarm optimization. Evolutionary Intelligence 12(2), 131–46 (2019). https://doi.org/10.1007/s12065-019-00199-5
122. Benaddi, H, Ibrahimi, K, Benslimane, A (eds.): Improving the intrusion detection system for NSL-KDD dataset based on PCA-fuzzy clustering-KNN 2018 6th International Conference on Wireless Networks and Mobile Communications (WINCOM). IEEE (2018)
123. Vinayakumar, R, Alazab, M, Soman, K, Poornachandran, P, Al-Nemrat, A, Venkatraman, S: Deep learning approach for intelligent intrusion detection system. Ieee Access 7, 41525–50 (2019). https://doi.org/10.1109/access.2019.2895334
124. Zhang, W, Han, D, Li, K-C, Massetto, FI: Wireless sensor network intrusion detection system based on MK-ELM. Soft Computing 24(16), 1–14 (2020). https://doi.org/10.1007/s00500-020-04678-1
125. Tang, Y, Gu, L, Wang, L: Deep Stacking Network for Intrusion Detection. Sensors 22(1), 25 (2022). https://doi.org/10.3390/s22010025
126. Al-Yaseen, WL, Idrees, AK, Almasoudy, FH: Wrapper Feature Selection Method based Differential Evolution and Extreme Learning Machine for Intrusion Detection System. Pattern Recognition 132, 108912 (2022). https://doi.org/10.1016/j.patcog.2022.108912
127. Gao, X, Shan, C, Hu, C, Niu, Z, Liu, Z: An adaptive ensemble machine learning model for intrusion detection. IEEE Access 7, 82512–21 (2019). https://doi.org/10.1109/access.2019.2923640
128. Wu, K, Chen, Z, Li, W: A novel intrusion detection model for a massive network using convolutional neural networks. Ieee Access 6, 50850–9 (2018). https://doi.org/10.1109/access.2018.2868993
129. Ding, Y, Zhai, Y (eds.): Intrusion detection system for NSL-KDD dataset using convolutional neural networks Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence (2018)
130. Bedi, P, Gupta, N, Jindal, V: I-SiamIDS: an improved Siam-IDS for handling class imbalance in network-based intrusion detection systems. Applied Intelligence 51(2), 1133–51 (2021). https://doi.org/10.1007/s10489-020-01886-y
131. Ayubkhan, SAH, Yap, W-S, Morris, E, Rawthar, MBK: A practical intrusion detection system based on denoising autoencoder and LightGBM classifier with improved detection performance. Journal of Ambient Intelligence and Humanized Computing 14(6), 1–26 (2022). https://doi.org/10.1007/s12652-022-04449-w
132. Gu, Z, Wang, L, Li, J, Wen, M, Liu, Y: Intrusion Detection Method Based on Stacked Sparse Autoencoder and Sliced GRU for Connected Healthcare Systems. Arabian Journal for Science and Engineering 48(2), 1–14 (2022). https://doi.org/10.1007/s13369-022-07079-8
133. De Carvalho, Bertoli G, Junior, LAP, Saotome, O, dos Santos, AL: Generalizing intrusion detection for heterogeneous networks: A stacked-unsupervised federated learning approach. Computers & Security 127, 103106 (2023). https://doi.org/10.1016/j.cose.2023.103106
134. Wang, C, Sun, Y, Lv, S, Wang, C, Liu, H, Wang, B: Intrusion Detection System Based on One-Class Support Vector Machine and Gaussian Mixture Model. Electronics 12(4), 930 (2023). https://doi.org/10.3390/electronics12040930
135. Muhammad, G, Hossain, MS, Garg, S: Stacked autoencoder-based intrusion detection system to combat financial fraudulent. IEEE Internet of Things Journal 10(3), 2071–2078 (2020). https://doi.org/10.1109/jiot.2020.3041184
136. Ortega-Fernandez, I, Sestelo, M, Burguillo, JC, Piñón-Blanco, C: Network intrusion detection system for DDoS attacks in ICS using deep autoencoders. Wireless Networks 30(6), 1–17 (2023). https://doi.org/10.1007/s11276-022-03214-3
137. Vu, L, Nguyen, QU, Nguyen, DN, Hoang, DT, Dutkiewicz, E: Deep generative learning models for cloud intrusion detection systems. IEEE Transactions on Cybernetics 53(1), 565–77 (2022). https://doi.org/10.1109/tcyb.2022.3163811
138. Kalpana, R: Recurrent nonsymmetric deep auto encoder approach for network intrusion detection system. Measurement: Sensors 24, 100527 (2022). https://doi.org/10.1016/j.measen.2022.100527
139. DAS, A, PRAMOD, S: AN ENHANCED OPTIMIZATION MODEL WITH ENSEMBLE AUTOENCODER FOR ZERO-DAY ATTACK DETECTION. Journal of Theoretical and Applied Information Technology 100(22) (2022)
140. Alissa, KA, Alotaibi, SS, Alrayes, FS, Aljebreen, M, Alazwari, S, Alshahrani, H, Ahmed Elfaki, M., Othman, M., Motwakel, A.: Crystal Structure Optimization with Deep-Autoencoder-Based Intrusion Detection for Secure Internet of Drones Environment. Drones 6(10), 297 (2022). https://doi.org/10.3390/drones6100297
141. Khanam, S, Ahmedy, I, Idris, MYI, Jaward, MH: Towards an Effective Intrusion Detection Model Using Focal Loss Variational Autoencoder for Internet of Things (IoT). Sensors 22(15), 5822 (2022). https://doi.org/10.3390/s22155822
142. Neuschmied, H, Winter, M, Stojanović, B, Hofer-Schmitz, K, Božić, J, Kleb, U: APT-Attack Detection Based on Multi-Stage Autoencoders. Applied Sciences 12(13), 6816 (2022). https://doi.org/10.3390/app12136816
143. Li, Z, Chen, S, Dai, H, Xu, D, Chu, C-K, Xiao, B: Abnormal Traffic Detection: Traffic Feature Extraction and DAE-GAN With Efficient Data Augmentation. IEEE Transactions on Reliability 72(2), 498–510 (2022). https://doi.org/10.1109/tr.2022.3204349
144. Wang, W, Du, X, Shan, D, Qin, R, Wang, N: Cloud intrusion detection method based on stacked contractive auto-encoder and support vector machine. IEEE transactions on cloud computing 10(3), 1634–46 (2020). https://doi.org/10.1109/tcc.2020.3001017
145. Badji, JCJ, Diallo, C (eds.): A CNN-based Attack Classification versus an AE-based Unsupervised Anomaly Detection for Intrusion Detection Systems 2022 International Conference on Electrical, Computer and Energy Technologies (ICECET). IEEE (2022)
146. He, J, Wang, X, Song, Y, Xiang, Q, Chen, C: Network intrusion detection based on conditional wasserstein variational autoencoder with generative adversarial network and one-dimensional convolutional neural networks. Applied Intelligence 53(10), 1–21 (2022). https://doi.org/10.1007/s10489-022-03995-2
147. Sumathi, S, Rajesh, R, Lim, S: Recurrent and Deep Learning Neural Network Models for DDoS Attack Detection. Journal of Sensors 2022, 1–21 (2022). https://doi.org/10.1155/2022/8530312
148. Ketepalli, G, Bulla, P (eds.): Feature Extraction using LSTM Autoencoder in Network Intrusion Detection System 2022 7th International Conference on Communication and Electronics Systems (ICCES). IEEE (2022)
149. Chikkalwar, SR, Garapati, Y: Autoencoder-support vector machine-grasshopper optimization for intrusion detection system. Int J Intell Eng Syst. 15(4), 406–14 (2020)
150. Wang, C, Liu, H, Sun, Y, Wei, Y, Wang, K, Wang, B: Dimension Reduction Technique Based on Supervised Autoencoder for Intrusion Detection of Industrial Control Systems. Security and Communication Networks 2022, 1–12 (2022). https://doi.org/10.1155/2022/5713074
151. Mhamdi, L, Isa, MM: Securing SDN: Hybrid autoencoder-random forest for intrusion detection and attack mitigation. Journal of Network and Computer Applications 225, 103868 (2024). https://doi.org/10.1016/j.jnca.2024.103868
152. Le, T-D, Truong, HBH, Kim, D: Multi-classification in-vehicle intrusion detection system using packet-and sequence-level characteristics from time-embedded transformer with autoencoder. Knowledge-Based Systems 299, 112091 (2024). https://doi.org/10.1016/j.knosys.2024.112091
153. Bi, J, Guan, Z, Yuan, H, Zhang, J: Improved network intrusion classification with attention-assisted bidirectional LSTM and optimized sparse contractive autoencoders. Expert Systems with Applications 244, 122966 (2024). https://doi.org/10.1016/j.eswa.2023.122966
154. Khaw, YM, Jahromi, AA, Arani, MF, Kundur, D: Evasive attacks against autoencoder-based cyberattack detection systems in power systems. Energy and AI 17, 100381 (2024). https://doi.org/10.1016/j.egyai.2024.100381
155. Tahir, M, Abdullah, A, Udzir, NI, Kasmiran, KA: A novel approach for handling missing data to enhance network intrusion detection system. Cyber Security and Applications 3, 100063 (2025). https://doi.org/10.1016/j.csa.2024.100063

156. Hore, S, Ghadermazi, J, Shah, A, Bastian, ND: A sequential deep learning framework for a robust and resilient network intrusion detection system. Computers & Security 144, 103928 (2024). https://doi.org/10.1016/j.cose.2024.103928
157. Nixon, C, Sedky, M, Champion, J, Hassan, M: SALAD: A split active learning based unsupervised network data stream anomaly detection method using autoencoders. Expert Systems with Applications 248, 123439 (2024). https://doi.org/10.1016/j.eswa.2024.123439
158. Shrestha, R, Mohammadi, M, Sinaei, S, Salcines, A, Pampliega, D, Clemente, R, Sanz, A.L., Nowroozi, E., Lindgren, A.: Anomaly detection based on lstm and autoencoders using federated learning in smart electric grid. Journal of Parallel and Distributed Computing 193, 104951 (2024). https://doi.org/10.1016/j.jpdc.2024.104951
159. Long, C, Xiao, J, Wei, J, Zhao, J, Wan, W, Du, G (eds.): Autoencoder ensembles for network intrusion detection 2022 24th International Conference on Advanced Communication Technology (ICACT). IEEE (2022)
160. Rao, KN, Rao, KV, PVGD, PR: A hybrid intrusion detection system based on sparse autoencoder and deep neural network. Computer Communications 180, 77–88 (2021). https://doi.org/10.1016/j.comcom.2021.08.026
161. Binbusayyis, A, Vaiyapuri, T: Unsupervised deep learning approach for network intrusion detection combining convolutional autoencoder and one-class SVM. Applied Intelligence 51(10), 7094–108 (2021). https://doi.org/10.1007/s10489-021-02205-9
162. Duhayyim, MA, Alissa, KA, Alrayes, FS, Alotaibi, SS, Tag El Din, EM, Abdelmageed, AA, Yaseen, I., Motwakel, A.: Evolutionary-Based Deep Stacked Autoencoder for Intrusion Detection in a Cloud-Based Cyber-Physical System. Applied Sciences 12(14), 6875 (2022). https://doi.org/10.3390/app12146875
163. Pandey, JK, Kumar, S, Lamin, M, Gupta, S, Dubey, RK, Sammy, F: A Metaheuristic Autoencoder Deep Learning Model for Intrusion Detector System. Mathematical Problems in Engineering 2022, 1–11 (2022). https://doi.org/10.1155/2022/3859155
164. Artur, M: Review the performance of the Bernoulli Naïve Bayes Classifier in Intrusion Detection Systems using Recursive Feature Elimination with Cross-validated selection of the best number of features. Procedia Computer Science 190, 564–70 (2021). https://doi.org/10.1016/j.procs.2021.06.066
165. Khammassi, C, Krichen, S: A GA-LR wrapper approach for feature selection in network intrusion detection. computers & security 70, 255–77 (2017). https://doi.org/10.1016/j.cose.2017.06.005
166. Dissanayake, K, Md Johar, MG: Comparative study on heart disease prediction using feature selection techniques on classification algorithms. Applied Computational Intelligence and Soft Computing 2021, 1–17 (2021). https://doi.org/10.1155/2021/5581806
167. Jain, D, Singh, V: Feature selection and classification systems for chronic disease prediction: A review. Egyptian Informatics Journal. 19(3), 179–89 (2018). https://doi.org/10.1016/j.eij.2018.03.002
168. Khraisat, A, Gondal, I, Vamplew, P, Kamruzzaman, J: Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity 2(1), 1–22 (2019). https://doi.org/10.1186/s42400-019-0038-7
169. Mera-Gaona, M, López, DM, Vargas-Canas, R, Neumann, U: Framework for the ensemble of feature selection methods. Applied Sciences 11(17), 8122 (2021). https://doi.org/10.3390/app11178122
170. Patil, R, Dudeja, H, Gawade, S, Modi, C (eds.): Protocol specific multi-threaded network intrusion detection system (PM-NIDS) for DoS/DDoS attack detection in cloud 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE (2018)
171. Al-Fawa'reh, M, Al-Fayoumi, M, Nashwan, S, Fraihat, S: Cyber threat intelligence using PCA-DNN model to detect abnormal network behavior. Egyptian Informatics Journal 23(2), 173–85 (2022). https://doi.org/10.1016/j.eij.2021.12.001
172. Chakraborty, D, Verma, AK, Sharma, S, Bhakar, R (eds.): Feature Selection based False Data Detection Scheme using Machine Learning for Power System 2022 IEEE Bombay Section Signature Conference (IBSSC). IEEE (2022)
173. Di Mauro, M, Galatro, G, Fortino, G, Liotta, A: Supervised feature selection techniques in network intrusion detection: A critical review. Engineering Applications of Artificial Intelligence 101, 104216 (2021). https://doi.org/10.1016/j.engappai.2021.104216
174. Ravi, V, Chaganti, R, Alazab, M: Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. Computers and Electrical Engineering 102, 108156 (2022). https://doi.org/10.1016/j.compeleceng.2022.108156
175. Alam, F, Kashef, R, Jaseemuddin, M (eds.): Enhancing the performance of network traffic classification methods using efficient feature selection models 2021 IEEE International Systems Conference (SysCon). IEEE (2021)
176. Boquet, G, Morell, A, Serrano, J, Vicario, JL: A variational autoencoder solution for road traffic forecasting systems: Missing data imputation, dimension reduction, model selection and anomaly detection. Transportation Research Part C: Emerging Technologies 115, 102622 (2020). https://doi.org/10.1016/j.trc.2020.102622
177. D'Angelo, G, Palmieri, F: Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction. Journal of Network and Computer Applications 173, 102890 (2021). https://doi.org/10.1016/j.jnca.2020.102890
178. Setitra, MA, Fan, M, Bensalem, ZEA: An efficient approach to detect distributed denial of service attacks for software defined internet of things combining autoencoder and extreme gradient boosting with feature selection and hyperparameter tuning optimization. Transactions on Emerging Telecommunications Technologies 34(9), e4827 (2023). https://doi.org/10.1002/ett.4827
179. Xu, W, Jang-Jaccard, J, Singh, A, Wei, Y, Sabrina, F: Improving performance of autoencoder-based network anomaly detection on nsl-kdd dataset. IEEE Access 9, 140136–46 (2021). https://doi.org/10.1109/access.2021.3116612
180. Khraisat, A, Alazab, A: A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges. Cybersecurity 4, 1–27 (2021). https://doi.org/10.1186/s42400-021-00077-7
181. Moualla, S, Khorzom, K, Jafar, A: Improving the Performance of Machine Learning-Based Network Intrusion Detection Systems on the UNSW-NB15 Dataset. Computational Intelligence and Neuroscience 2021(1), 5557577 (2021). https://doi.org/10.1155/2021/5557577
182. Yang, Z, Liu, X, Li, T, Wu, D, Wang, J, Zhao, Y, Han, H.: A systematic literature review of methods and datasets for anomaly-based network intrusion detection. Computers & Security 116, 102675 (2022). https://doi.org/10.1016/j.cose.2022.102675
183. Bakro, M, Kumar, RR, Alabrah, A, Ashraf, Z, Ahmed, MN, Shameem, M, Abdelsalam, A.: An improved design for a cloud intrusion detection system using hybrid features selection approach with ML classifier. IEEE Access 11, 64228–64247 (2023). https://doi.org/10.1109/access.2023.3289405
184. Nazir, A, Khan, RA: A novel combinatorial optimization based feature selection method for network intrusion detection. Computers & Security 102, 102164 (2021). https://doi.org/10.1016/j.cose.2020.102164
185. Thakkar, A, Lohiya, R: A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. Artificial Intelligence Review 55(1), 453–563 (2022). https://doi.org/10.1007/s10462-021-10037-9
186. Kocher, G, Kumar, G: Machine learning and deep learning methods for intrusion detection systems: recent developments and challenges. Soft Computing 25(15), 9731–63 (2021). https://doi.org/10.1007/s00500-021-05893-0
187. Lin, H, Xue, Q, Feng, J, Bai, D: Internet of things intrusion detection model and algorithm based on cloud computing and multi-feature extraction extreme learning machine. Digital Communications and Networks 9(1), 111–24 (2023). https://doi.org/10.1016/j.dcan.2022.09.021
188. Rahman, MA, Asyhari, AT, Leong, L, Satrya, G, Tao, MH, Zolkipli, M: Scalable machine learning-based intrusion detection system for IoT-enabled smart cities. Sustainable Cities and Society 61, 102324 (2020). https://doi.org/10.1016/j.scs.2020.102324

189. Subbiah, S, Anbananthen, KSM, Thangaraj, S, Kannan, S, Chelliah, D: Intrusion detection technique in wireless sensor network using grid search random forest with Boruta feature selection algorithm. Journal of Communications and Networks 24(2), 264–73 (2022). https://doi.org/10.23919/jcn.2022.000002
190. Ding, H, Chen, L, Dong, L, Fu, Z, Cui, X: Imbalanced data classification: A KNN and generative adversarial networks-based hybrid approach for intrusion detection. Future Generation Computer Systems 131, 240–54 (2022). https://doi.org/10.1016/j.future.2022.01.026
191. Park, S, Park, H: Combined oversampling and undersampling method based on slow-start algorithm for imbalanced network traffic. Computing 103(3), 401–24 (2021). https://doi.org/10.1007/s00607-020-00854-1
192. Wang, Z-M, Tian, J-Y, Qin, J, Fang, H, Chen, L-M: A Few-Shot Learning-Based Siamese Capsule Network for Intrusion Detection with Imbalanced Training Data. Computational intelligence and neuroscience 2021(1), 7126913 (2021). https://doi.org/10.1155/2021/7126913
193. Dina, AS, Siddique, A, Manivannan, D: Effect of balancing data using synthetic data on the performance of machine learning classifiers for intrusion detection in computer networks. IEEE Access 10, 96731–47 (2022). https://doi.org/10.1109/access.2022.3205337
194. Harini, R, Maheswari, N, Ganapathy, S, Sivagami, M: An effective technique for detecting minority attacks in NIDS using deep learning and sampling approach. Alexandria Engineering Journal 78, 469–82 (2023). https://doi.org/10.1016/j.aej.2023.07.063
195. Shin, Y, Kim, M, Kim, H: Towards unbalanced multiclass intrusion detection with hybrid sampling methods and ensemble classification. Applied Soft Computing 157, 111517 (2024). https://doi.org/10.1016/j.asoc.2024.111517
196. Alotaibi, SD, Yadav, K, Aledaily, AN, Alkwai, LM, Yousef Dafhalla, AK, Almansour, S, Lingamuthu, V.: Deep Neural Network-Based Intrusion Detection System through PCA. Mathematical Problems in Engineering 2022(1), 6488571–9 (2022). https://doi.org/10.1155/2022/6488571
197. Rajadurai, H, Gandhi, UD: An empirical model in intrusion detection systems using principal component analysis and deep learning models. Computational Intelligence 37(3), 1111–24 (2021). https://doi.org/10.1111/coin.12342
198. Andresini, G, Appice, A, Malerba, D: Autoencoder-based deep metric learning for network intrusion detection. Information Sciences 569, 706–27 (2021). https://doi.org/10.1016/j.ins.2021.05.016
199. Prabu, S, Padmanabhan, J, Bala, G (eds.): Effective ensemble dimensionality reduction approach using denoising autoencoder for intrusion detection system Intelligent Sustainable Systems: Proceedings of ICISS 2021. Springer (2022)
200. Wang, C, Liu, H, Sun, Y, Wei, Y, Wang, K, Wang, B: Dimension reduction technique based on supervised autoencoder for intrusion detection of industrial control systems. Security and Communication Networks 2022(1), 5713074–12 (2022). https://doi.org/10.1155/2022/5713074
201. Sahu, A, Wlazlo, P, Mao, Z, Huang, H, Goulart, A, Davis, K, Zonouz, S.: Design and evaluation of a cyber-physical testbed for improving attack resilience of power systems. IET Cyber-Physical Systems: Theory & Applications 6(4), 208–27 (2021). https://doi.org/10.1049/cps2.12018
202. Robinson, B: The Communications Security Establishment (CSE). Top Secret Canada: Understanding the Canadian Intelligence and National Security Community, pp. 72–89 (2021)
203. Alsumaini, AYM: Two-Stage Ensemble Learning for NIDS Multiclass Classification. Hamad Bin Khalifa University, Qatar (2023)
204. Bagui, S, Kalaimannan, E, Bagui, S, Nandi, D, Pinto, A: Using machine learning techniques to identify rare cyber-attacks on the UNSW-NB15 dataset. Security and Privacy 2(6), e91 (2019). https://doi.org/10.1002/spy2.91
205. Zhang, L, Xie, X, Xiao, K, Bai, W, Liu, K, Dong, P: MANomaly: Mutual adversarial networks for semi-supervised anomaly detection. Information Sciences 611, 65–80 (2022). https://doi.org/10.1016/j.ins.2022.08.033
206. Almomani, I, Al-Kasasbeh, B, Al-Akhras, M: WSN-DS: A dataset for intrusion detection systems in wireless sensor networks. Journal of Sensors 2016, 1–16 (2016). https://doi.org/10.1155/2016/4731953
207. Farhat, S, Abdelkader, M, Meddeb-Makhlouf, A, Zarai, F (eds.): Comparative study of classification algorithms for cloud ids using nsl-kdd dataset in weka 2020 International Wireless Communications and Mobile Computing (IWCMC). IEEE (2020)
208. Zhang, C, Jia, D, Wang, L, Wang, W, Liu, F, Yang, A: Comparative research on network intrusion detection methods based on machine learning. Computers & Security 121, 102861 (2022). https://doi.org/10.1016/j.cose.2022.102861
209. Boutaba, R, Salahuddin, MA, Limam, N, Ayoubi, S, Shahriar, N, Estrada-Solano, F, Caicedo, O.M.: A comprehensive survey on machine learning for networking: evolution, applications and research opportunities. Journal of Internet Services and Applications 9(1), 1–99 (2018). https://doi.org/10.1186/s13174-018-0087-2
210. Badillo, S, Banfai, B, Birzele, F, Davydov, II, Hutchinson, L, Kam-Thong, T, Siebourg-Polster, J., Steiert, B., Zhang, J.D.: An introduction to machine learning. Clinical pharmacology & therapeutics 107(4), 871–85 (2020). https://doi.org/10.1002/cpt.1796
211. O'Mahony, N, Campbell, S, Carvalho, A, Harapanahalli, S, Hernandez, GV, Krpalkova, L, et al. (eds.) Deep learning vs. traditional computer vision. Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), vol. 1 1. Springer (2020)
212. Maulud, D, Abdulazeez, AM: A review on linear regression comprehensive in machine learning. Journal of Applied Science and Technology Trends 1(4), 140–7 (2020). https://doi.org/10.38094/jastt1457
213. Rath, S, Tripathy, A, Tripathy, AR: Prediction of new active cases of coronavirus disease (COVID-19) pandemic using multiple linear regression model. Diabetes & Metabolic Syndrome: Clinical Research & Reviews 14(5), 1467–74 (2020). https://doi.org/10.1016/j.dsx.2020.07.045
214. Wahab, L, Jiang, H: A comparative study on machine learning based algorithms for prediction of motorcycle crash severity. PLoS One 14(4), e0214966 (2019). https://doi.org/10.1371/journal.pone.0214966
215. Adiat, K, Akeredolu, B, Akinlalu, A, Olayanju, G: Application of logistic regression analysis in prediction of groundwater vulnerability in gold mining environment: a case of Ilesa gold mining area, southwestern, Nigeria. Environmental Monitoring and Assessment 192(9), 1–17 (2020). https://doi.org/10.1007/s10661-020-08532-7
216. Rezapour, M, Molan, AM, Ksaibati, K: Analyzing injury severity of motorcycle at-fault crashes using machine learning techniques, decision tree and logistic regression models. International journal of transportation science and technology 9(2), 89–99 (2020). https://doi.org/10.1016/j.ijtst.2019.10.002
217. Sarhan, M, Layeghy, S, Moustafa, N, Gallagher, M, Portmann, M: Feature extraction for machine learning-based intrusion detection in IoT networks. Digital Communications and Networks 10(1), 205–216 (2022). https://doi.org/10.1016/j.dcan.2022.08.012
218. Aghaei, S, Azizi, MJ, Vayanos, P (eds.): Learning optimal and fair decision trees for non-discriminative decision-making Proceedings of the AAAI conference on artificial intelligence (2019)
219. Sarker, IH, Colman, A, Han, J, Khan, AI, Abushark, YB, Salah, K: Behavdt: a behavioral decision tree learning to build user-centric context-aware predictive model. Mobile Networks and Applications 25(3), 1151–61 (2020). https://doi.org/10.1007/s11036-019-01443-z
220. Galal, MA, Hussein, WM, El-din abdel Kawy, E, Sayed, MM (eds.) Satellite battery fault detection using Naïve Bayesian classifier. IEEE Aerospace Conference; IEEE (2019)
221. Zhao, X, Wei, H, Wang, H, Zhu, T, Zhang, K: 3D-CNN-based feature extraction of ground-based cloud images for direct normal irradiance prediction. Solar Energy 181, 510–8 (2019). https://doi.org/10.1016/j.solener.2019.01.096
222. Boyko, N, Boksho, K (eds.) Application of the Naive Bayesian Classifier in Work on Sentimental Analysis of Medical Data. IDDM (2020)
223. Uddin, S, Haque, I, Lu, H, Moni, MA, Gide, E: Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its

different variants for disease prediction. Scientific Reports 12(1), 1–11 (2022). https://doi.org/10.1038/s41598-022-10358-x
224. Pascual-Triana, JD, Charte, D, Andrés Arroyo, M, Fernández, A, Herrera, F: Revisiting data complexity metrics based on morphology for overlap and imbalance: snapshot, new overlap number of balls metrics and singular problems prospect. Knowledge and Information Systems 63(7), 1961–89 (2021). https://doi.org/10.1007/s10115-021-01577-1
225. Sen, PC, Hajra, M, Ghosh, M (eds.): Supervised classification algorithms in machine learning: A survey and review Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018. Springer (2020)
226. Cervantes, J, Garcia-Lamont, F, Rodríguez-Mazahua, L, Lopez, A: A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing 408, 189–215 (2020). https://doi.org/10.1016/j.neucom.2019.10.118
227. Liu, H, Lang, B: Machine learning and deep learning methods for intrusion detection systems: A survey. applied sciences 9(20), 4396 (2019). https://doi.org/10.3390/app9204396
228. Nabipour, M, Nayyeri, P, Jabani, H, Shahab, S, Mosavi, A: Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis. IEEE Access 8, 150199–212 (2020). https://doi.org/10.1109/access.2020.3015966
229. Ganaie, MA, Hu, M, Malik, A, Tanveer, M, Suganthan, P: Ensemble deep learning: A review. Engineering Applications of Artificial Intelligence 115, 105151 (2022). https://doi.org/10.1016/j.engappai.2022.105151
230. Qutub, A, Al-Mehmadi, A, Al-Hssan, M, Aljohani, R, Alghamdi, HS: Prediction of employee attrition using machine learning and ensemble methods. Int J Mach Learn Comput. 11(2), 110–4 (2021). https://doi.org/10.18178/ijmlc.2021.11.2.1022
231. Frank, M, Drikakis, D, Charissis, V: Machine-learning methods for computational science and engineering. Computation 8(1), 15 (2020). https://doi.org/10.3390/computation8010015
232. Vaish, J, Datta, SS (eds.): Short-term load forecasting using bootstrap aggregation based ensemble method 2021 7th International Conference on Electrical Energy Systems (ICEES). IEEE (2021)
233. Farsi, M: Application of ensemble RNN deep neural network to the fall detection through IoT environment. Alexandria Engineering Journal 60(1), 199–211 (2021). https://doi.org/10.1016/j.aej.2020.06.056
234. Sagi, O, Rokach, L: Ensemble learning: A survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8(4), e1249 (2018). https://doi.org/10.1002/widm.1249
235. Zhang, W, Ramezani, R, Naeim, A (eds.): WOTBoost: Weighted oversampling technique in boosting for imbalanced learning 2019 IEEE International Conference on Big Data (Big Data). IEEE (2019)
236. Ghatasheh, N, Faris, H, Abukhurma, R, Castillo, PA, Al-Madi, N, Mora, AM, Al-Zoubi, A.M., Hassanat, A.: Cost-sensitive ensemble methods for bankruptcy prediction in a highly imbalanced data distribution: A real case from the Spanish market. Progress in Artificial Intelligence 9(4), 361–75 (2020). https://doi.org/10.1007/s13748-020-00219-x
237. Shah, K, Patel, H, Sanghvi, D, Shah, M: A comparative analysis of logistic regression, random forest and KNN models for the text classification. Augmented Human Research 5, 1–16 (2020). https://doi.org/10.1007/s41133-020-00032-0
238. Jiang, M, Liu, J, Zhang, L, Liu, C: An improved Stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms. Physica A: Statistical Mechanics and its Applications 541, 122272 (2020). https://doi.org/10.1016/j.physa.2019.122272
239. Fath-allah K, Abdelghani B. Object Detection with Convolutional Neural Networks: faculté des sciences et de la technologie univ bba; 2022
240. Shrestha, A, Mahmood, A: Review of deep learning algorithms and architectures. IEEE access 7, 53040–65 (2019). https://doi.org/10.1109/access.2019.2912200
241. Dixit, P, Silakari, S: Deep learning algorithms for cybersecurity applications: A technological and status review. Computer Science Review 39, 100317 (2021). https://doi.org/10.1016/j.cosrev.2020.100317
242. Kamath, U, Liu, J, Whitaker, J: Deep learning for NLP and speech recognition. Springer (2019)
243. Côté-Allard, U, Fall, CL, Drouin, A, Campeau-Lecours, A, Gosselin, C, Glette, K, Laviolette, F., Gosselin, B.: Deep learning for electromyographic hand gesture signal classification using transfer learning. IEEE transactions on neural systems and rehabilitation engineering 27(4), 760–71 (2019). https://doi.org/10.1109/tnsre.2019.2896269
244. Li, D, Deng, L, Gupta, BB, Wang, H, Choi, C: A novel CNN based security guaranteed image watermarking generation scenario for smart city applications. Information Sciences 479, 432–47 (2019). https://doi.org/10.1016/j.ins.2018.02.060
245. Chang, Y, Chen, J, Qu, C, Pan, T: Intelligent fault diagnosis of wind turbines via a deep learning network using parallel convolution layers with multi-scale kernels. Renewable Energy 153, 205–13 (2020). https://doi.org/10.1016/j.renene.2020.02.004
246. Cao, X, Liu, J, Meng, F, Yan, B, Zheng, H, Su, H (eds.): Anomaly detection for screw tightening timing data with LSTM recurrent neural network 2019 15th International Conference on Mobile Ad-Hoc and Sensor Networks (MSN). IEEE (2019)
247. Liu, S, Mallol-Ragolta, A, Parada-Cabaleiro, E, Qian, K, Jing, X, Kathan, A, Hu, B., Schuller, B.W.: Audio self-supervised learning: A survey. Patterns 3(12), 100616 (2022). https://doi.org/10.1016/j.patter.2022.100616
248. Goodfellow, I, Pouget-Abadie, J, Mirza, M, Xu, B, Warde-Farley, D, Ozair, S, Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM 63(11), 139–44 (2020). https://doi.org/10.1145/3422622
249. Clark, K, Luong, M-T, Le, QV, Manning, CD: Electra: Pre-training text encoders as discriminators rather than generators. arXiv preprint arXiv:200310555 (2020)
250. Chorowski, J, Weiss, RJ, Bengio, S, Van Den Oord, A: Unsupervised speech representation learning using wavenet autoencoders. IEEE/ACM transactions on audio, speech, and language processing 27(12), 2041–53 (2019). https://doi.org/10.1109/taslp.2019.2938863
251. Xia, M, Li, T, Liu, L, Xu, L, de Silva, CW: Intelligent fault diagnosis approach with unsupervised feature learning by stacked denoising autoencoder. IET Science, Measurement & Technology 11(6), 687–95 (2017). https://doi.org/10.1049/iet-smt.2016.0423
252. Pathirage, CSN, Li, J, Li, L, Hao, H, Liu, W, Ni, P: Structural damage identification based on autoencoder neural networks and deep learning. Engineering structures 172, 13–28 (2018). https://doi.org/10.1016/j.engstruct.2018.05.109
253. Liu, H-I, Chen, W-L: Re-transformer: a self-attention based model for machine translation. Procedia Computer Science 189, 3–10 (2021). https://doi.org/10.1016/j.procs.2021.05.065
254. Chang, Y, Li, F, Chen, J, Liu, Y, Li, Z: Efficient temporal flow Transformer accompanied with multi-head probsparse self-attention mechanism for remaining useful life prognostics. Reliability Engineering & System Safety 226, 108701 (2022). https://doi.org/10.1016/j.ress.2022.108701
255. Bolbot, V, Kulkarni, K, Brunou, P, Banda, OV, Musharraf, M: Developments and research directions in maritime cybersecurity: A systematic literature review and bibliometric analysis. International Journal of Critical Infrastructure Protection 39, 100571 (2022). https://doi.org/10.1016/j.ijcip.2022.100571
256. Altarturi, HH, Saadoon, M, Anuar, NB: Cyber parental control: A bibliometric study. Children and Youth Services Review 116, 105134 (2020). https://doi.org/10.1016/j.childyouth.2020.105134
257. Debrah, C, Chan, AP, Darko, A: Artificial intelligence in green building. Automation in Construction 137, 104192 (2022). https://doi.org/10.1016/j.autcon.2022.104192
258. Raza, B., et al.: Performance prediction and adaptation for database management system workload using case-based reasoning approach. Information Systems 76(1), 46–58 (2018). https://doi.org/10.1016/j.is.2018.04.005
259. Butt, R.A., et al.: A survey of dynamic bandwidth assignment schemes for TDM-based passive optical network. Journal of Optical

Communications 41(3), 279–293 (2020). https://doi.org/10.1515/joc-2017-0186
260. Raza, B., et al.: Autonomic performance prediction framework for data warehouse queries using lazy learning approach. Applied Soft Computing 91(4), 106216 (2020). https://doi.org/10.1016/j.asoc.2020.106216
261. Faheem, M., et al.: A multiobjective, lion mating optimization inspired routing protocol for wireless body area sensor network based healthcare applications. Sensors 19(23), 5072 (2019). https://doi.org/10.3390/s19235072
262. Faheem, M., et al.: A blockchain-based resilient andsecure framework for events monitoring and control indistributed renewable energy systems. IET Blockchain, 1–15 (2024). https://doi.org/10.1049/blc2.12081
263. Al-Khasawneh, M.A.S., et al.: A MapReduce based approach for secure batch satellite image encryption. IEEE Access 11(5), 62865–62878 (2023). https://doi.org/10.1109/ACCESS.2023.3279719
264. Khan, A.A., et al.: D2PAM: epileptic seizures prediction using adversarial deep dual patch attention mechanism. CAAI Transactions on Intelligence Technology 8(3), 755–769 (2023). https://doi.org/10.1049/cit2.12261
265. Faheem, M, et al.: A lightweight smartcontracts framework for blockchain-based securecommunication in smart grid applications. IET Gener. Transm. Distrib 18(3), 625–638 (2024). https://doi.org/10.1049/gtd2.13103
266. Akram, A., et al.: Segmentation and classification of skin lesions using hybrid deep learning method in the Internet of Medical Things. Skin Research and Technology 29(11), e13524 (2023). https://doi.org/10.1111/srt.13524
267. Faheem, M., Al-Khasawneh, M.A.: Multilayer cyberattacks identification and classification using machine learning in internet of blockchain (IoBC)-based energy networks. Data in Brief 54(3), 110461 (2024). https://doi.org/10.1016/j.dib.2024.110461
268. Faheem, M., et al.: Cyberattack patterns in blockchain-based communication networks for distributed renewable energy systems: a study on big datasets. Data in Brief 55(4), 110212 (2024). https://doi.org/10.1016/j.dib.2024.110212

How to cite this article: John, A., et al.: Intrusion detection in cluster-based wireless sensor networks: Current issues, opportunities and future research directions. IET Wirel. Sens. Syst. 14(6), 293–332 (2024). https://doi.org/10.1049/wss2.12100