




Article

Anomaly Detection in 6G Networks Using Machine Learning Methods

Mamoon M. Saeed ¹, Rashid A. Saeed ² , Maha Abdelhaq ^{3,*}, Raed Alsaqour ^{4,*} , Mohammad Kamrul Hasan ⁵  and Rania A. Mokhtar ²

- ¹ Department of Communications and Electronics Engineering, Faculty of Engineering, University of Modern Sciences (UMS), Baghdad Street, Sanaa 20031, Yemen; mamoon530@gmail.com
- ² Department of Computer Engineering, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia; eng_rashid@hotmail.com (R.A.S.); ramohammed@tu.edu.sa (R.A.M.)
- ³ Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
- ⁴ Department of Information Technology, College of Computing and Informatics, Saudi Electronic University, P.O. Box 93499, Riyadh 11673, Saudi Arabia
- ⁵ Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), Bangi 43600, Malaysia; hasankamrul@ieee.org
- * Correspondence: msabdelhaq@pnu.edu.sa (M.A.); raed.ftsm@gmail.com (R.A.)

Abstract: While the cloudification of networks with a micro-services-oriented design is a well-known feature of 5G, the 6G era of networks is closely related to intelligent network orchestration and management. Consequently, artificial intelligence (AI), machine learning (ML), and deep learning (DL) have a big part to play in the 6G paradigm that is being imagined. Future end-to-end automation of networks requires proactive threat detection, the use of clever mitigation strategies, and confirmation that 6G networks will be self-sustaining. To strengthen and consolidate the role of AI in safeguarding 6G networks, this article explores how AI may be employed in 6G security. In order to achieve this, a novel anomaly detection system for 6G networks (AD6GNs) based on ensemble learning (EL) for communication networks was redeveloped in this study. The first stage in the EL-ADCN process is pre-processing. The second stage is the feature selection approach. It applies the reimplemented hybrid approach using a comparison of the ensemble learning and feature selection random forest algorithms (CFS-RF). NB2015, CIC_IDS2017, NSL KDD, and CICDDOS2019 are the three datasets, each given a reduced dimensionality, and the top subset characteristic for each is determined separately. Hybrid EL techniques are used in the third step to find intrusions. The average voting methodology is employed as an aggregation method, and two classifiers—support vector machines (SVM) and random forests (RF)—are modified to be used as EL algorithms for bagging and adaboosting, respectively. Testing the concept of the last step involves employing classification forms that are binary and multi-class. The best experimental results were obtained by applying 30, 35, 40, and 40 features of the reimplemented system to the three datasets: NSL_KDD, UNSW_NB2015, CIC_IDS2017, and CICDDOS2019. For the NSL_KDD dataset, the accuracy was 99.5% with a false alarm rate of 0.0038; the accuracy was 99.9% for the UNSW_NB2015 dataset with a false alarm rate of 0.0076; and the accuracy was 99.8% for the CIC_IDS2017 dataset with a false alarm rate of 0.0009. However, the accuracy was 99.95426% for the CICDDOS2019 dataset, with a false alarm rate of 0.00113.

Keywords: anomaly detection; 6G networks; machine learning; ensemble approach; bagging algorithm; correlation feature selection algorithm



Citation: Saeed, M.M.; Saeed, R.A.; Abdelhaq, M.; Alsaqour, R.; Hasan, M.K.; Mokhtar, R.A. Anomaly Detection in 6G Networks Using Machine Learning Methods. *Electronics* **2023**, *12*, 3300. <https://doi.org/10.3390/electronics12153300>

Academic Editors: Christos J. Bouras and Juan-Carlos Cano

Received: 6 June 2023

Revised: 17 July 2023

Accepted: 28 July 2023

Published: 31 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

After 2030, users will engage with internet virtual worlds through 6G communications [1]. The 5G network's intended digital transformation has already begun and will

develop over the next ten years [2–5]. After 2030, networks will need to be equipped with cutting-edge technologies that allow connected intelligence in digital virtual worlds in order to address networking and communication issues. According to published research, 6G systems will have new application fields, such as wireless brain–computer interfaces (BCI), networked robotics and autonomous systems, and extended multimodal reality (XR) applications [6]. At the same time, traditional applications, such as multimedia streaming, will continue to exist. Other 6G use cases [7] that demand high data throughput, incredibly low latencies, and incredibly reliable networks include holographic telepresence, eHealth, and in-body networks.

Identifying zero-day attacks is a difficult issue. Many suspicious actions are discovered every day. The effects of these complex intrusions can develop into significant dangers that make it more challenging for current intrusion detection systems (IDSs) [8–14]. IDSs send out alerts when they find unusual activity or an identified threat. Every negative act that compromises the information system is considered an incursion [15]. IDSs watch computer systems for any unusual activity that a typical packet filter would miss. They look for indicators of potentially dangerous behavior, cyber resistance to disruptive activities, and unauthorized access to the system in network packets. The two techniques employed by IDSs to identify intrusions are anomaly intrusion detection systems (AIDS) and signature intrusion detection systems (SIDS) [16]. AIDS is flawed, and it has a high percentage of false alarms. A unique IDS model was offered to overcome these issues in order to increase accuracy and decrease FAR. This model incorporates both SIDS and AIDS. In contrast to AIDS, SIDS might identify frequent intrusions [17].

Data mining techniques are employed in intrusion detection (ID), a data analysis method, to automatically find, extract, and differentiate between normal and invasive patterns. DM problems typically fall into one of four categories: classification, grouping, regression, or association rule learning. A crucial step in the IDS process is the feature selection (FS) technique, which identifies the important features and eliminates the useless ones to reduce performance degradation [18]. The correlation-based heuristic assessment function used by correlation FS (CFS) allows users to prioritize features. It contrasts the attribute vector subsets that are tied to class labels but not to one another. The CFS algorithm mandates excluding features that have little relevance to the class. Due to their frequent connections to one or more of the other aspects, excess features should be looked into [19].

Classification, grouping, regression, and association rule learning are the four general areas in which DM problems can be categorized. The FS technique, a critical stage in the IDS process, detects the critical features and eliminates the unimportant ones to prevent performance degradation [20]. Users can order characteristics according to their importance using CFS's correlation-based heuristic assessment function. It contrasts attribute vector subsets connected to class labels but not to one another. Features barely relevant to the class must be excluded, according to the CFS algorithm. Excessive features should be investigated due to their frequent linkages to one or more of the other aspects [21].

Ingenuous mitigation techniques, proactive threat identification, and self-sustaining networks in 6G are all required to automate forthcoming networks. Determining and responding to possible attacks based on network abnormalities rather than cryptographic approaches requires an end-to-end security strategy that uses AI techniques. The security requirements and difficulties for 6G applications are listed in Table 1. This research will investigate how AI can help 6G networks be more secure.

Table 1. Security issues with 6G apps and the fundamental security requirement.

References	Challenges in Security	6G Application	Requirements for Security
[22,23]	<ul style="list-style-type: none"> Resources are scarce. Energy is at a premium. Design of a complete security system. 	Holographic telepresence	<ul style="list-style-type: none"> Superior privacy. Operation in real-time. Stopping terrorist assaults.

Table 1. Cont.

References	Challenges in Security	6G Application	Requirements for Security
[24,25]	<ul style="list-style-type: none"> High altitude, high movement, limited energy, and a variety of equipment are all factors. Attacks by terrorists. Physical interference. A lack of security regulations. 	Mobility based on UAVs	<ul style="list-style-type: none"> A range of gadgets. Real-time operations with lower operating expenses. High scalability. Design of an end-to-end security system.
[26]	<ul style="list-style-type: none"> Attacks that include physical meddling; scarcity of resources. 	Prolonged reality	<ul style="list-style-type: none"> A security edges. Minimal privacy. Operation in real-time.
[27–29]	<ul style="list-style-type: none"> Highly mobile. Physical assaults. Privacy issues. End-to-end security that is lightweight. A range of gadgets. Innovative security measures. 	Autonomous connected vehicles	<ul style="list-style-type: none"> Lightweight authentication that is also ultra-private. Preventative security. Immediate defense against assaults. Low communication and computation costs.
[30,31]	<ul style="list-style-type: none"> Smart grid intrusions. Data aggregation. Translation of one protocol into another. Attacks using actual equipment. Exploitation. 	Smart grid 2.0	<ul style="list-style-type: none"> Heterogeneity and scalable IoT security. Security with no touches. Excellent privacy. Cost savings. Ensuring access.
[32–34]	<ul style="list-style-type: none"> Denial of service. Supply chain and extended systems, smart security, smart factories, and so on. 	Industry 5.0	<ul style="list-style-type: none"> Extremely high privacy. Preventative security. Small-scale security. Intellectual property and secret information.
[35]	<ul style="list-style-type: none"> Innovative techniques for dynamic security. A range of gadgets. Trustworthiness. Visibility. Ethical and legal considerations. Viability and extensibility. Managed security operations. 	Healthcare using artificial intelligence	<ul style="list-style-type: none"> A range of gadgets. Superior privacy. Zero-touch protection. Security at the edge. Domain-specific security is available.
[36,37]	<ul style="list-style-type: none"> Double spending, majority vulnerability, and double-spending. Scalability. A quantum computer. Leakage of transaction privacy. 	Distributed ledger software	<ul style="list-style-type: none"> Avoiding double-spending attacks and privacy breaches

Table 1. Cont.

References	Challenges in Security	6G Application	Requirements for Security
[38–40]	<ul style="list-style-type: none"> Security of the digital model as well as the physical model. A range of gadgets. Privacy-preserving. Very good mobility. Separate security measures. 	Virtual twins.	<ul style="list-style-type: none"> Broadband use. Superb privacy. Secure lightly. Scalability. Flexible security measures. Robustness.
[41–43]	<ul style="list-style-type: none"> Design of the building; physical assaults. Privacy issues. Systems for total security. 	Brain–computer connections that are wireless.	<ul style="list-style-type: none"> Accessibility Integrity Safety Confidentiality.

The world of technology is shifting toward hardware and simple algorithms, IoE, and complex networking. Although there are ongoing studies, intrusion detection systems are unable to optimize their detection rate (DR), false alarm rate (FAR), false negative rate (FNR), false positive rate (FPR), or execution time due to the huge dimensions of the standard dataset and the incidence of zero-day assaults. Even though it directly affects resources, time complexity has not been acknowledged as a significant component. In order to extract the best subset of the original features, dimensionality reduction with an FS is reimplemented in this study. The accuracy and stability of the intrusion detection systems are then increased while minimizing the necessary calculation time by forwarding the reimplemented hybrid EL to these subsets. The reimplemented process develops hybrid EL and FS algorithms to obtain precise and effective IDs.

In the reimplemented approach, an anomaly message is detected and classified as spam or ham using novel AD6GNs based on EL for wireless communication networks. ML approaches substantially contribute to detecting spam tweets, images/videos, and SMS messages on mobile devices. An IDS guards computer networks against malicious invasions and is used to check for network weaknesses. For network analysis, the three main classifications of an intrusion detection system are signature-based, anomaly-based, and hybrid-based. ML methods significantly aid in detecting various intrusions on host and network systems. The main contributions of this research are as follows:

1. Within the FS framework, a unique CFS-RF method is utilized to evaluate the correlation of the chosen features. Improving the efficiency of the testing and training processes is quite advantageous.
2. The effectiveness of the multi-class and binary forms will be enhanced and used on the three unbalanced datasets. The process explains hybrid ensemble algorithms by modifying two different classifiers to function as adaboosting, followed by the average voting technique (bagging method) to combine judgments from several classification judgments using an ensemble of classifiers, such as SVM and RF.
3. This article investigates how AI may be used in 6G security to strengthen and reinforce safeguarding 6G networks. This study reimplements a novel AD6GN using EL to do this.

The remainder of this paper is organized as follows: Many similar works are presented in Section 2. In Section 3, the basics of threats and attacks are discussed. At the same time, Section 4 AI/ML technology in the 6G network is discussed. Section 5 offers a thorough explanation of the methodology and the reimplemented system. Section 6 discusses various ML techniques, shows how the reimplemented system was put into practice using the datasets that were used, and presents the discussion and analysis of the results of this and other studies. Section 7 ends with a summary of the findings and recommendations for additional investigation.

2. Related Work

While research on the security of sixth-generation (6G) wireless communication systems has already started to give smooth connectivity to the growing number of users and to improve ML services, the 5G network systems are still in the process of being fully standardized. The problem of system secrecy rate optimization has already been extensively studied using a variety of contemporary methodologies. The authors in [44] used an IRS-assisted SWIPT system with a user-equipment (UE)-installed power-splitting (PS) scheme to study the physical-layer security and transmission optimization issues. It is crucial to jointly determine the lowest collected energy and maximum transmitter power required to maximize the system secrecy rate, as well as the transmitter power, the UE's PS factor, and the IRS's phase shift matrix [45]. They suggested utilizing an alternating optimization (AO)-based approach to get the optimum answers.

The authors' major goal in [46] was to provide the most recent developments in DL-based physical layer techniques in order to open the door for exciting 6G applications. In [47], the authors outline their vision for scalable, reliable edge AI systems that merge wireless communication plans with distributed machine learning models. An intrusion detection system for networks, an ensemble based on decision trees and rule learners, was the idea put forth by the authors in [48]. Using the NSL KDD dataset for IDSs with erratic base classifiers, a unique DAR ensemble architecture was suggested. For accuracy, DR, and FAR, respectively, the experimental findings revealed 80%, 81%, and 15.1%.

The authors in [49] used KNN certainty factor voting classifiers, linear discriminant analysis, and ML-based two-class classification models to minimize dimensionality. The network imbalance of the anomaly datasets was addressed using the SMOTE approach. The model was trained using two newly constructed training datasets. When 16 characteristics were selected, the experimental NSL-KDD evaluation revealed an accuracy of 83.24%, a FAR of 4.83%, a TPR of 82%, and a FPR of 5.43.

According to the authors in [50], we used a feature subset based on correlation, information gain, symmetrical uncertainty, and NSL-KDD datasets utilizing GAR-Forest, which stands for greedy randomized adaptive search strategies. Information gain reached an accuracy of 78.9035% with 10 features for multi-class, while the results indicated an accuracy of 85.0559% with 32 features for binary class.

The authors in [51] suggested using NSL KDD and recognizing attacks on wireless sensor networks. To this end, "anomaly detection in hybrid wireless sensor networks and machine learning techniques for energy efficiency" were used. According to the experimental findings, precision, re-call, and F1-Score were 94.00%, 98.00%, and 96.00%, respectively, while accuracy was 95%.

The authors in [52] recommended the FS approach based on genetic algorithms and logistic regression for network-based intrusion detection systems (NIDS) based on EL algorithms. The findings for CIC_IDS2017, NSL_KDD, and UNSW_NB2015, utilizing 11, 8, and 13 features, respectively, demonstrated 98.99%, 98.73%, and 97.997% accuracy with 98.75%, 96.64%, and 98.93% detection rates.

They introduced the CSE-IDS, which uses cost-sensitive DL and ensemble approaches to address an unbalanced class of intrusion detection systems. Phase 1 separates and disperses routine or suspicious network attacks using a deep neural network (DNN). There are three phases that must be discussed. Extreme gradient boosting categorizes phase 2 primary attacks, whereas phase 3 adopts RF to classify minor attacks. They used the datasets NSL_KDD, CIDDs 001, and CIC_IDS2017 to assess the performance of the system; the accuracy for NSL, CIDDs-001, and CIC_IDS2017, respectively, was 99%, 96%, and 92%, although the complexity time measurement took several hours [53].

The researchers used these datasets to evaluate the system's performance. To extract features, the authors used a stacked auto-encoding network [54]. They then suggested a random forest, SVM, and another classification technique. Using a hybrid DNN-KNN methodology, the authors in [55] provided a binary classification. There were several ML and DL hybrid algorithms. In [56], authors employed ensemble learning approaches and

feature selection (i.e., correlation feature selection-forest attribute). The study's experimental findings exclusively utilized the CIC IDS2017 dataset. Additionally, when employing 30 feature-selected tests, the testing accuracy was 87%. Using just two datasets—NSL-KDD and UNSW BN2015—the authors of [57] introduced correlation feature selection methods. These approaches allow users to choose the optimal feature. Furthermore, they only selected those datasets 30 features.

By utilizing an ensemble learning (EL) algorithm-based anomaly detection in communication networks (ADCNs), the authors of [58] suggested a unique anomaly detection in communication networks. In each of the three datasets (NSL_KDD, UNSW_NB2015, and CIC_IDS2017) individually, it minimizes dimensionality and finds the best subset feature. Their results were obtained by applying 30, 35, and 40 features of the system to the three datasets, which resulted in accuracy values of 99.6% for NSL_KDD, 99.1% for UNSW_NB2015, and 99.4% for CIC_IDS2017. Since the work in [58] was the primary study, we applied the same research methods and reimplemented its algorithms for the purpose of benchmarking with a variety of dataset pre-processing and feature combinations. In addition to that, we utilized the more recent dataset known as CICDDOS2019 and concentrated our attention on data anomalies in 6G cellular networks.

For 5G networks, the authors of [59] offered a special feature extraction and detection system where the deep belief network (DBN) and bidirectional long-short-term memory (Bi-LSTM) were utilized as part of a hybrid classifier (HC). The weights of both Bi-LSTM and DBN are optimized using a novel deer hunting updated sun flower optimization (DHSFO) model, which combines the concepts of sun flower optimization (SFO) and deer hunting optimization (DHO) methods, to transform the detecting stage properly and precisely.

A unique architecture was put forth by the developers of [60] with the goal of proactively detecting such traffic anomalies or forecasting impending 5G network traffic anomalies before they happen. Clustering and decision tree-based learning are applied to automatically identify anomalies in 5G network traffic. For the pro-active identification of impending network traffic behaviors, a time series model—more specifically, a Bidirectional long short-term memory (BiLSTM) autoencoder—is used. According to evaluation results, the proposed framework is effective and viable and has the potential to be more successful than a state-of-the-art solution, with a prediction accuracy of up to 90.02%.

Using a deep-learning methodology, the authors of [61] proposed a new technique for anomaly identification that is based on the autoencoder principle and long-short-term memory (LSTM) recurrent neural networks. Real sensor data was used in a number of simulations, and the accuracy of the discovered abnormalities was assessed. The findings indicate that the RNNs have an accuracy of 87% in detecting anomalies over both short and extended periods of time. SDWSNs and RNNs can therefore be utilized as a first step in effectively detecting different kinds of anomalies.

Many researchers have examined distributed machine learning techniques [62,63] and found that they significantly reduce the complexity of high dimensional data. These researchers have demonstrated the advantages of utilizing ML algorithms to handle enormous amounts of data for the IDS pre-processing step. However, DL algorithms could access hidden features in the step that classified anomalies from many targets to find unidentified attacks. They achieve the highest rates of false alarm, false negative, and detection, and they greatly outperform state-of-the-art technologies while consuming the least time when applied to many datasets.

To address computing cost and privacy preservation issues, the authors of [64] introduced a synonym-based multi-keyword ranked search over encrypted cloud data. To allocate an index vector for each document, the suggested mechanism builds an m-Way search tree. It then uses a depth-first search to compute the top score ranking, improving search efficiency. The examination's findings demonstrate that the presumed index vectors might be separated into sub-vectors before being stored in the index tree to improve computing efficiency and security.

3. Threats and Attacks in 6G Networks

Although 6G networks are still being developed, it is crucial to take into account any potential dangers and assaults that might occur in these networks. Here are a few potential 6G network dangers and assaults [65,66]:

- **Malware:** Much like in current mobile networks, malware is projected to pose a serious danger to 6G networks. Malware can steal confidential data, interfere with network operations, or harm network infrastructure.
- **DDoS assaults:** Distributed denial of service (DDoS) attacks are a typical way to interfere with a network's functionality. DDoS attacks involve the use of several devices to flood a network with traffic, making it difficult or impossible for legitimate traffic to pass through.
- **IoT botnets:** It is anticipated that 6G networks will support a significant number of IoT devices, which are susceptible to attacks. IoT devices have the potential to be exploited to build botnets, which may then be used to execute DDoS assaults or other kinds of attacks.
- **Rogue base stations:** These devices can intercept and alter network traffic or attack other network-connected devices.
- **Spoofing attacks:** An attacker can use spoofing attacks, which pose as legitimate networks or devices, to obtain sensitive information or initiate attacks on other network-connected devices.
- **Eavesdropping:** Eavesdropping attacks can be used to intercept and steal sensitive information that is transferred over the network. Personal data, financial information, and other sensitive data types may be included.
- **Physical assaults:** Physical assaults on network infrastructure, such as cutting fiber optic cables or tampering with power sources, can impair network performance and do serious harm.
- **Man-in-the-middle (MITM) attacks:** These attacks allow an attacker to intercept and change network traffic in order to steal sensitive data or launch attacks on other network-connected devices.
- **Supply chain attacks:** By focusing on the vendors and suppliers who provide the network's components, supply chain attacks can be used to compromise network infrastructure.
- **Quantum assaults:** These attacks could risk 6G networks by breaking encryption and other currently thought-to-be-secure security measures using quantum computers.

Network operators and device makers must install strong security mechanisms, such as encryption, authentication, and access control, to handle these dangers and attacks in 6G networks. In order to identify and stop attacks, network operators may also need to invest in extra equipment, such as intrusion detection systems and firewalls. Additionally, it is critical to ensure that the 6G ecosystem's various parts—including devices, networks, and applications—are built with security in mind and routinely patched to address emerging threats and vulnerabilities.

To ensure that they are resilient to a variety of risks and attacks, 6G networks will ultimately need to be designed with a "security-first" approach. In addition to the continued research and development of new security technologies and procedures, this will call for tight cooperation between network operators, device manufacturers, and other 6G ecosystem partners [67,68].

Combining two detection methods results in hybrid-based detection. Denial of service (DoS), remote to local (R2L), user to root (U2R), and probing are four different types of cyberattacks. U2R attacks occur when users attempt to obtain root or admin user access rights. R2L attacks occur when a remote user attempts to log in as a local user. DoS refers to the phenomenon where a genuine user is prevented from accessing the system by a hostile actor who is overloading the network resources. However, fraudsters scan the network when probing to identify vulnerable points for upcoming assaults.

4. AI/ML Technology in 6G Network

According to recent studies, the network architecture for all 6G network technologies must include AI and ML. The 6G networking industry has paid close attention to artificial intelligence. A lot of training data and strong processing capacity are needed to adopt AI/ML in 5G networks. However, AI/ML have become essential to 6G networks. Different frames of 6G's security defense and protection are secured using AI and ML. Security systems have improved in autonomy, accuracy, and predictiveness due to the use of AI and ML.

This subsection discusses a few of the issues with AI/ML in the 6G system [69,70]:

1. *Reliability*: AI manages network security and the accuracy of machine learning models and components.
2. *Visibility*: Real-time AI and ML-based security functions are monitored to assure credibility and control.
3. *Ethical and Legal Considerations*: AI-based optimization approaches can potentially exclude specific clients or applications. Whether or not AI-powered security solutions protect all users, AI controls who oversees security services that fail.
4. *Versatility and adaptability*: Protected data transfers are essential to protecting federated learners' privacy. AI/ML struggle with the scalability of the necessary computation, storage resources, and communication.
5. *Managed security duties*: There could be a lot of overhead when AI/ML security solutions are linked to huge data activities. The learning and inference phases should be safe and secure for the model's flexibility. The anticipated intelligent 6G system would leverage advanced AI methods and methodologies to satisfy the demands of new use cases, high service needs, and necessary capabilities. Figure 1 shows the AI/ML-based 6G secured architecture; the following sentences characterize it.

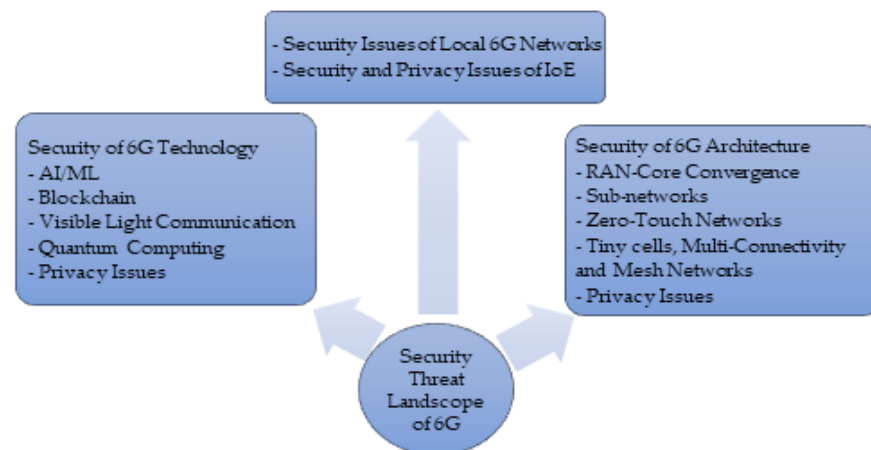


Figure 1. Attacks on each 6G AI/ML security architecture tier are distinct.

4.1. Security Improvements for 5G

In comparison to earlier generations of mobile networks, 5G networks feature significant security enhancements, but they also present new security difficulties. The following are a few of the major security upgrades in 5G networks [71–74]:

1. *Better encryption*: 5G networks employ better encryption methods to secure data in transit. The advanced encryption standard (AES) with 256-bit keys, which is regarded as being very secure, is required by the 5G standard.
2. *Network slicing*: 5G networks divide the network into virtual networks, each with its own set of security guidelines and restrictions. This makes it possible to separate and secure various types of traffic using various levels of protection.

3. User authentication: To defend against identity theft and other sorts of threats, 5G networks use better user authentication protocols, including 5G authentication and key agreement (AKA).
4. Secure boot: Secure boot certifies that the operating system and software of 5G devices are valid and have not been tampered with.
5. SIM-based security: SIM-based security is used by 5G networks to thwart SIM swapping attacks, in which criminals attempt to seize control of a user's SIM card to obtain access to their account.
6. Software-defined networking: Software-defined networking (SDN) enables dynamic network configuration and security policies to be updated in real-time to address new threats and assaults; 5G networks use SDN to enable this functionality.
7. Network function virtualization: Network function virtualization (NFV) is used in 5G networks to enable the implementation of network functions in software instead of hardware. This can lessen the danger of hardware-based attacks and make updating and securing network operations easier.
8. Improved IoT security: To guard against assaults on IoT devices, 5G networks include enhanced security mechanisms for IoT devices, including mutual authentication and secure communication protocols.
9. Improved privacy: 5G networks include improved privacy features such as the capacity to send data in a manner that conceals the user's identity, making it harder for attackers to monitor or identify users.
10. Better network management: 5G networks now have greater granular control over network resources and security regulations thanks to network slicing and SDN solutions.

While 5G networks have significant security advantages over earlier mobile network generations, they also present new security difficulties, such as the increased use of edge computing and the requirement for security measures to be implemented on a variety of devices. To solve these issues and guarantee the security of 5G networks, network operators, device makers, and other players in the 5G ecosystem must continue to develop and adopt new security measures. This will require continued innovation and cooperation, as well as a "security-first" mindset in the development of 5G networks and devices [72].

4.2. ML Technology in 6G Network

As ML technology may be used to enhance network performance, optimize resource allocation, and enable new services and applications, it is anticipated to play a significant role in 6G networks. Listed below are a few potential uses for ML technology in 6G networks [73–77]:

1. Network optimization: To enhance network performance and lower latency, ML algorithms can be used to evaluate network traffic and optimize network resources, such as bandwidth and power. In order to prevent network slowdowns, ML can also be used to forecast network congestion and modify network resources in real time.
2. Intelligent resource allocation: ML algorithms can be used to intelligently allocate network resources, such as frequency bands and spectra, to increase network capacity and performance. This can be crucial in 6G networks, which are anticipated to accommodate a variety of devices and applications with various bandwidth and latency requirements.
3. Anomaly detection: ML algorithms can be used to identify unexpected data transmission patterns in network traffic, which may indicate security breaches or other problems. ML can enable quicker responses to security threats and other network concerns by detecting real-time anomalies.
4. Predictive maintenance: ML algorithms can be used to forecast the need for maintenance on base stations and antennas, two types of network infrastructure. By analyzing sensor and other data, ML algorithms can find patterns that could point to potential problems or maintenance requirements. This can enable preventative upkeep and repair, minimizing downtime and enhancing network efficiency.

5. Intelligent edge computing: ML algorithms can be installed on edge devices such as gateways or routers to enable real-time data processing and analysis at the network edge. As a result, less data will need to be sent across long distances, enhancing network efficiency and lowering latency. Additionally, by optimizing edge computing resources such as processing speed and memory, ML can make data processing more effective and efficient.
6. Intelligent network management: ML algorithms can be used to examine data on network performance to pinpoint areas that can be improved and to streamline network management procedures. ML can be used, for instance, to forecast network issues and suggest proactive maintenance, or it can be used to optimize network routing to ease congestion and boost performance.
7. Enhanced security: By quickly identifying and addressing security issues, ML algorithms can improve network security. For instance, ML can be used to recognize strange devices or traffic patterns on a network and automatically activate security measures in response.

In general, ML technology can boost network security in 6G networks, enable new services and applications, and enhance network performance [78]. However, implementing machine learning algorithms in 6G networks will necessitate a substantial investment in infrastructure for artificial intelligence and machine learning and careful consideration of data privacy and security issues [79–83]. Additionally, in order to ensure interoperability and compatibility amongst various components of the 6G ecosystem, the deployment of ML in 6G networks would necessitate close cooperation between network operators, device makers, and application developers [84–86].

5. Research Methodology

The reimplemented architecture has several processes for detecting anomalies. The first component of the defensive system is an intrusion detection system with databases hidden behind the firewall (network data is preprocessed). Following preprocessing, it is necessary to identify any replacement of the null values with alternate values after the system checks for any missing values. Default consideration is given to average values, after which duplicate values are purged from the dataset. Dimensional reduction is applied to the encoded data to facilitate data management. Anomaly detection is aided by feature optimization, which is performed to extract the best characteristics from the data.

The filtered data is then moved on to the next stage, where the method known as CFS-RF is used only to choose the affected features for the outcomes. In order to distinguish between legitimate activity and potential attacks, the system employs the reimplemented hybrid adaboosting bagging algorithms (HABBAs) as classifiers. The system's intricate construction is shown in Figure 2.

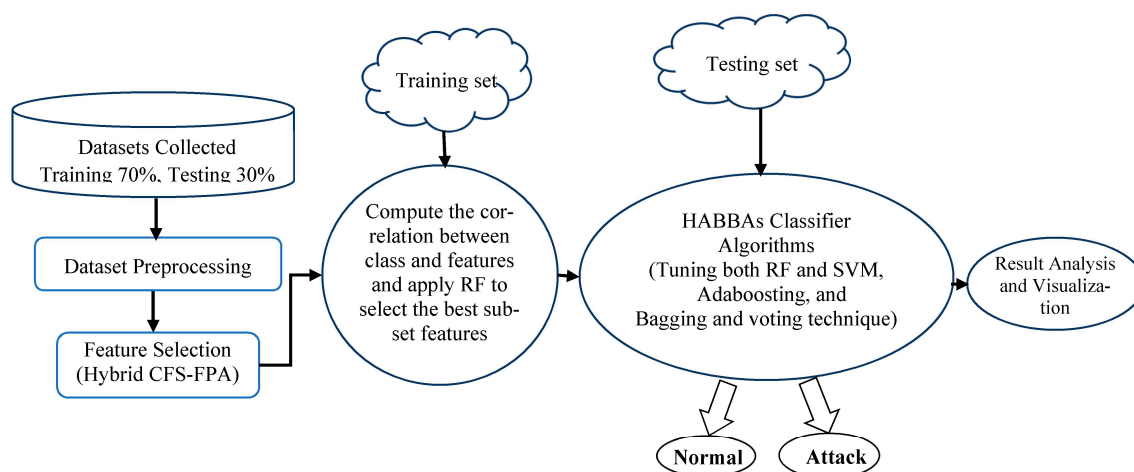


Figure 2. The anomaly detecting system's construction.

It comprises several phases, each of which completes a particular task in a series of sequential stages. The previous step serves as input for the following stage. These phases and steps are thoroughly discussed. After reading the collected NSL_KDD, UNSW_NB2015, CIC_IDS2017, and CICDDOS2019 datasets, the three primary processes of the preprocessing stage (filtration, transformation, and normalization) are carried out.

5.1. Simulation Steps

Here are the steps we used to simulate a new anomaly detection approach based on group learning to detect attacks in 6G communications using Python with multiple datasets:

- We define the scope and goals of our simulation. Before we start running the simulation, we work on defining the scope of the simulation and the goals we want to achieve. Which included defining the types of attacks we want to detect, the performance metrics we want to improve, and the datasets we want to use.
- Collect and process the datasets: For our simulation experiment, we will need to collect and preprocess three datasets: NB2015, CIC_IDS2017, CICDDOS2019, and NSL KDD, which involves cleaning and pre-processing the data to remove any noise or outliers and converting the data into a format suitable for machine learning and deep learning. Which also needs to divide the data into training, validation, and test sets.
- Implement a group learning approach: the next step is to implement a group learning approach to detect anomalies in the connectivity networks, which involves selecting suitable basic models, such as decision trees, neural networks, or support vector machines, and combining them using an appropriate clustering method, such as packing, boosting, or stacking. Python libraries such as scikit-Learn or TensorFlow were used to implement and train these models.
- Train and validate the model: Once the group learning approach has been implemented, the model can be trained and validated using the three datasets, which may include splitting data sets into training, validation, and test sets and using cross-validation to evaluate model performance. Measures such as accuracy, recall, and F1 score were used to evaluate the model's performance.
- Improving the model: After validating the model, its performance was improved by adjusting the hyperparameters, choosing different basic models, using different clustering methods, and integrating the equations.
- Visualizing and interpreting the results: After refining the model, the results were visualized and interpreted to gain insights into the attack detection problem and the model's performance. This involved creating visualizations of the data and the model and interpreting the results in the context of the specific attack detection problem.

5.2. Description of Datasets

This system implements experiments using the datasets from the UNSW NB2015, the NSL KDD, the CIC_IDS2017, and the CICDDOS2019. The initial data collection is NSL-KDD. It was developed as a strong preprocessing stage (the first step of the reimplemented system) to decrease the forecasting challenges. Different baseline classifiers were used to categorize records with five different levels of complexity. Next to each occurrence, the number of accurate predictions was noted [86–88].

The proportion of records picked for each difficulty degree category in the first KDD-Cup'99 dataset is inversely related to the number of records chosen. In our sample, we used 125.973 occurrences of the KDD Train set, which included 58.631 attacks and 67.342 instances of routine traffic. Many current low-key assaults were included in the second dataset (UNSW-NB15), which aimed to replicate current network configurations. It contained 45 columns (id = 1, features = 44), 2,540,045 records spread over four big-data CSV files, and records for testing a total of 82,334 and training a total of 175,342, respectively [89].

The CIC_IDS2017 contained current large attacks [87] and benign data in addition to the results of the network traffic analysis using the CIC flow meters. Flows were used to

time stamp protocols, IP addresses at the source and destination, ports, and all attacks. The most recent dataset also included updates for threats such as port scanning, DDoS, brute force, XSS, SQL injection, penetration, and botnets. A total of 2,830,744 records were spread among 8 files, each containing 78 unique named features [90].

5.3. Stages in Pre-Processing

The preprocessing converts the unprocessed data, in three steps, into an appropriate format for analysis before applying it. Algorithm 1 explains these processes (i.e., filtration, transformation, and normalization).

5.3.1. The Step of Filtration

Filtration removes unnecessary or useless material from the data, making the datasets simpler to use and comprehend. It distributes the remaining data and reorganizes it into classified groupings.

5.3.2. The Step of Transformation

Transformation converts a property's category value into a number using a one-hot encoding algorithm. For instance, it uses this function to transform several protocols, such as the transmission control protocol (TCP) and the user datagram protocol (UDP), into numerical data.

5.3.3. The Step of Normalization

The Minimax function is used to convert numbers between 0 and 1.

Algorithm 1 Datasets preprocessing

Input: Datasets D1, D2, D3, and D4 may be read. The letters D4, D3, D2 and D1 stand for, the CICDDOS2019, the CIC IDS2017, the UNSW NB2015 and the NSL.

Output : (Feature) Y_i Worth

Begin

1. Loop
2. Step 1 and Step 2:
3. **For** filtration
4. Do data transformation and
5. Remove occurrences that are pointless or unnecessary.
6. Distribution classification is set up.
7. **If** the input is not numeric, do
8. Statistics are obtained by applying the (One-Hot Encoding) function on categorical characteristics.
9. **End if**
10. **End For**
11. Step 3: Normative adjustment Using a minimum: Max = Determining the highest value. Min = Determining the Lowest Value.
12. $Y_i\text{Worth} = (Y_i\text{Worth} - \text{Min}) / (\text{Max} - \text{Min})$
13. Continue measures till all features have been completed.
14. Return Y_i Worth

End

5.4. CFS-RF Hybrid Method

CFS and bagging EL created a hybrid technique for effective FS and accurate categorization. As seen in Algorithm 2, the system takes advantage of the hybrid CFS-RF for FS that has been reimplemented. The following is a description of the reimplemented hybrid

CFS-RF method: It first uses the X_i value result from the preprocessing stage, then uses the merit equation to apply it to each feature as follows:

$$\mu_s = \frac{cr_k f}{\sqrt{c + c(c-1) + r_f f}} \quad (1)$$

where the correlation between features ($\overline{r_f f}$) is the correlation between features and classes ($\overline{r_k f}$). Equation (1) is explained by the calculated correlation (CFS). After that, it creates RF subsets using.

$$\{h(c, x, \theta_c = 1, 2, 3, \dots)\} \quad (2)$$

k stands for the integer, c represents the theta, h stands for frequency, and x represents a vector.

Algorithm 2 CFS-RF hybrid technique

Input: Features classes f1, f2, f3, and f4

Output: Final evaluation and estimated accuracy

Begin

1. Loop
2. **For** each merit of CFS generate a subset of random Forest Randomly
3. Do discriminate between the most relevant
4. and irrelevant features
5. **For** each subset
6. Compute w_i and σ^2
7. **End For**
8. Dimensionality reduction and selected most relevant features
9. **If** the datasets discretization, do
10. evaluation and estimated accuracy
11. **End if**
12. **End For**
13. Return Features classes

End

By computing the weight range, the redundant characteristics are verified:

$$wR^\lambda = \begin{cases} [0.000, (\frac{-1}{e^{-\lambda}})], \lambda = 1 \\ [(\frac{-1}{e^{\lambda-1}}) + p, e^{(\frac{-1}{\lambda})}], \lambda > 1 \end{cases} \quad (3)$$

where λ is the lambda, and wR^λ is the weight range. It determines the most pertinent characteristic with the least amount of standard deviation by calculating the standard division, σ_i , which is represented by:

$$\sigma_i = \frac{1.0 - \omega_i}{(n+1) - \lambda} \quad (4)$$

where the weight is represented by ω_i and standard division by σ_i .

5.5. Algorithms for Training, Testing, and Recognition Attack Using Hybrid Adaboosting and Bagging

In this phase, the hybrid EL algorithms are developed. To produce usable performance (RF and SVM), the following classifiers are tuned to run consecutively as adaboosting, using their updated weights.

5.5.1. Differentiated RF Classifier

The modified RF used for adaboosting is explained in Figure 3. The parameters and weights are also changed to improve the effectiveness of identifying unknown attacks. Initialization first equalizes all XiBest values with W_i and then uses an equation to produce RF subsets (5). Following that, it calculates the weight and standard deviation for each training set using:

$$p\sigma^2 + \frac{1-p}{B}\sigma^2 \quad (5)$$

where p stands for population and B is a constant. As a stopping condition in Figure 3, it is essential to calculate the value for each XiBest.

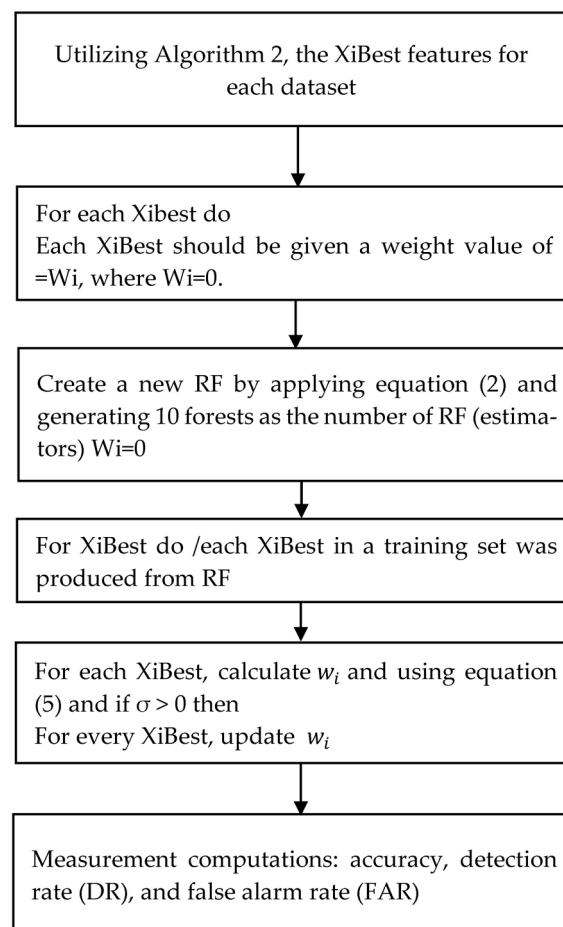


Figure 3. RF modified to function as AdaBoosting.

To obtain the best results from these updated classifiers, the reimplemented model aggregates them to work simultaneously as bagging and uses the weighted average voting strategy. The recommended HABBA's central idea is stated in Figure 3.

As a result of weight updates, Figure 3 modifies the RF to operate sequentially as adaboosting. Aggregation is applied to subsequently updated classifiers, voting using the weighted averages method to acquire the best results for variance and bias.

A modified version of this algorithm produces better results with lower error rates.

5.5.2. Classifier for Modified SVM

The dataset's weights for each XiBest feature are updated. The steps for creating the support vector, categorizing the datasets, and applying the improved SVM classifier are as follows:

$$f(x) = b + (x, w) \quad (6)$$

where weight (w) and bias (b) are two terms.

The adaboosting method is used in the first stage of Algorithm 3 with respect to every altered classifier (Figure 3). The value of each Classifier's weight is determined by:

$$(M_i) = \sum_{j=1}^d \text{err}(X_j) \times w_j \quad (7)$$

where $\text{err}(X_j)$ is the error for each classifier, and w_j is its weight. Afterward, by calculating the weight using and validating the error rate:

$$\log \frac{(1 - \text{ErrorRate}(M_i))}{\text{ErrorRate}(M_i)} \quad (8)$$

As a result, performance is improved and variation is reduced. The second step uses the bagging algorithm's basic idea to make these classifiers function as bootstraps. A shallower tree (during each phase of splitting), a sample of the variables, and a new dataset are utilized in the system to minimize overfitting, creating a composite model with reduced bias.

$$\text{Voting average} = \sum_{i=1}^1 \text{pci} \left(\frac{w_i}{x} \right) \quad (9)$$

The average voting method is calculated using Equation (9).

5.6. Confusion Matrix in Binary

Three datasets are used to implement the HABBAs. Each class, which has both normal and anomalous traffic, is manually applied to the confusion matrix. The given CFS-RF and HABBAs are submitted to the FSs (i.e., features 13, 30, 35, and 40) to detect intrusions. The confusion matrix is a kind of matrix often used in ML to evaluate the performance of algorithms. It summarizes the total correct and incorrect values predicted by the machine learning algorithms, as shown in Table 2.

Table 2. Confusion Matrix for Binary class.

	Class 1 Predicted	Class 2 Predicted
Class 1 actual	TP	FN
Class 2 actual	FP	TN

Details of the confusion matrix:

- Positive (P): Actually, positive.
- Negative (N): Actually, it is not positive.
- True Positive (TP): Actually positive and predicted as positive.
- False Negative (FN): Actually positive but predicted as negative.
- True Negative (TN): Actually negative and predicted as negative.
- False Positive (FP): Actually negative but predicted as positive.

The concept uses the binary class form of a confusion matrix. The NSL KDD classes are subject to suggestions. Tables 3 and 4, and Table 4 show the NSL KDD implementation using 30 FS as a binary class. In this FN, the four states are designated as true positive rate (TPR), false positive rate (FPR), true negative rate (TNR), and FNR. These four statuses affect the system's performance metrics. The tables demonstrate how the confusion matrices account for assault frequency and class distribution, demonstrating that the optimal combination of 30 attributes yields the best outcomes.

The NSL KDD confusion matrix for 30 features is shown in Table 3. The best FS is the 30-feature subset, as shown by Table 3's explanation of the NSL KDD binary type. For 35 features from the UNSW NB2015 dataset, see Table 4's confusion matrix. Forty features from the CIC IDS2017 dataset are listed in Table 5 as confusion matrices. The confusion matrix for 40 features from the CICDDOS2019 dataset is shown in Table 6.

Table 3. 30 features NSL_KDD confusion matrix.

Prediction		
	Benign	Attack
Benign	2886	0
Attack	1	9715

Table 4. Confusion matrix for 35 features from the UNSW_NB2015 dataset.

Prediction		
	Benign	Attack
Benign	746	0
Attack	27	1702

Algorithm 3 Combining HABBAs to detect intrusions

Begin

Training part:

1. Algorithms for AdaBoosting

Initialization: $k = 2$ (RF, SVM methods), $W_i = 0$ /weight of each class; The classifiers are C_i .2. Looping /* Equation (7) is used to determine M_i /, which stands for error rate, to modify classifiers (K_i) for each $i = 1$.3. If $M_i > 0.5$ then /* in the event that the error rate exceeds 0.5 */ For each procedure, Equation (8) is used to calculate (W_i). Compute each adjusted Classifier's prediction using $C_i = M_i$. W_i is added by C_i (i.e., RF and SVM) for every pair of classifiers.

4. End If

5. It is finished up until the two classifiers.

6. Return Minimum M_i ,

7. Algorithms for Bagging:

8. Looping /* SVM and RF are examples of modified classifiers C_i . */ applied the ensemble principle using a bootstrap model to each C_i . The voting concept is applied using Equation (9).

It is finished up until the two classifiers.

Test component: (Do the actions below)

Looping

To obtain both X_i -After and If X_i -Before, the accuracy for each (prediction) after and before is computed after voting.

9. Replacement process:

10. If X_i -Before is greater than X_i -After (averaging votes with the greatest likelihood).

11. else

The computation of FAR, FNR, DR, and accuracy

End else

12. End if

until the terminating requirement has been met.

13. Return Observations and a combined model.

End

Table 5. Confusion matrix for 40 features from the CIC_IDS2017 dataset.

Prediction		
	Benign	Attack
Benign	110,929	367
Attack	27	453,917

Different datasets are utilized to find new attacks, strengthening the system's defenses against outside threats (zero-day threats). The FNR and accuracy of these features are thoroughly explained in Table 7. Using 30 features in the reimplemented system can yield the best accuracy and FNR outcomes. FNR is the percentage of false negative discoveries

in an experiment separated by the detections of genuine positives and false negatives. Calculating the number of errors found for each assault that has been classified as normal is crucial for assessing the effectiveness and quality of the system.

Table 6. Confusion matrix for 40 features from the and CICDDOS2019 dataset.

		Prediction	
	Benign	Benign	Attack
	Attack	110,931 26	365 453,919

Table 7. HABBA accuracy by various FSs.

Datasets	Features Number	Accuracy	FNR	FN	TP
NSL_KDD	13	$9000 + 2280 / 12,600 = 0.89$	$605 / (605 + 9000) = 0.06$	605	9000
	30	$9714 + 2885 / 12,600 = 0.99$	$0 / (0 + 9714) = 0$	0	9714
	41	$9500 + 2480 / 12,600 = 0.95$	$405 / (405 + 9500) = 0.04$	405	9500
UNSW_NB2015	13	$1500 + 400 / 2470 = 0.76$	$344 / (344 + 1500) = 0.19$	344	1500
	35	$1701 + 744 / 2470 = 0.99$	$25 / (25 + 1701) = 0.01$	25	1701
	49	$1525 + 630 / 2470 = 0.87$	$201 / (201 + 1525) = 0.11$	201	1525
CIC_IDS2017	13	$492,176 / 565,562 = 0.87$	$62,736 / (62,736 + 443,615) = 0.123$	62,736	443,615
	40	$564,844 / 565,562 = 0.99$	$396 / (396 + 453,916) = 0.0008$	396	453,916
	78	$524,106 / 565,562 = 0.92$	$24,741 / (24,741 + 437,550) = 0.053$	24,741	4,377,550
CICDDOS2019	13	$492,180 / 572,302 = 0.86$	$72,634 / (72,634 + 543,813) = 0.1178$	72,634	543,813
	40	$564,921 / 565,203 = 0.9995$	$401 / (401 + 513,918) = 0.000779$	401	513,918
	78	$524,204 / 563,660 = 0.93$	$30,132 / (30,132 + 473,271) = 0.0598$	30,132	473,271
	91	$513,987 / 577,513 = 0.89$	$17,543 / (17,543 + 514,821) = 0.0329$	17,543	514,821

Additionally, the FNR and accuracy measures are insufficient for comparison when using 13 and 41 features. Also, displays the binary class for UNSW_NB2015, which has 35 features. Table 7 shows accuracy measurements for all the features used in these tables, including FN, TN, FP, and TP. It provides an explanation of the binary class of the UNSW_NB2015, demonstrating that the 35-feature FS is the ideal one to use for accurate attack distribution and diagnosis and normal traffic. 35 features are applied with a 0.01 FNR, resulting in a greater accuracy of 99%, as shown in Table 7's data.

The reimplemented system was evaluated across all FSs during CICDDOS2019, the last of the test sets. The binary class of the CICDDOS2019 confusion matrix with 40 characteristics. The TP, TN, FP, and FN accuracy measures are shown in Table 8 for all FSs and datasets. When using the reimplemented system with 40 features, the FNR of 0.00779 indicates the best accuracy of 99.95%.

Table 8. Attack F-score and accuracy when using the reimplemented system.

id	Attack Type	F-Score	Accuracy
0	DDoS	99%	98%
1	SQL injection	99%	100%
2	Brute Force	100%	98.4%
3	DoS Hulk	99%	99%
4	DoS Golden Eye	99%	98.9%
5	Botnet	100%	100%
6	DoS Slow Loris	99%	99%
7	FTP Patator	99%	98%
8	Web Attack	99%	100%
9	XSS	100%	100%
10	Infiltration	99%	99%

Applying 13 features results in the lowest accuracy of 86% and a FNR of 0.1178. Applying 78 features results in the highest accuracy of 93%, with an FNR of 0.0598. Applying 91 features results in the highest accuracy of 89%, with an FNR of 0.0329. Table 8 uses the f-score measure to show the precision of each assault in the dataset. The best results across all classes using the reimplemented system are shown in Table 8, with Brute Force, XXS, and Botnet achieving 100%. This suggests that the number of characteristics is the right one for recognizing all types of attacks.

6. Experimental Results and Discussion

6.1. System Configuration and Implementation

Three separate NSL KDD, UNSW NB 2015, CIC_IDS2017, and CICDDOS2019 datasets were utilized in implementing the reimplemented approach. The reimplemented system was tested using the remaining 30% of the datasets after 70% had been used for training. The reimplemented work was pushed through 3 sets of selected features—13, 30, 13, 35, and 13, 40, all employing CFS RF for the NSL KDD dataset, UNSW NB2015, CIC_IDS2017, and 13, 30 for the NSL KDD dataset, respectively—in order to gauge how well it performed. When employing CFS and collecting these s, it carried out RF with penalizing characteristics for these s, randomly selecting 10 estimators (10 subsets), and an ensemble.

Evaluation learning datasets compute correlations between features. It calculates W_i for each set by using the highest weight and disregarding inferior weights. In the end, only the most influential subset of features—those that impact the performance of intrusion detection—were chosen. The dataset's dimensionality was decreased, and unnecessary attributes were removed using the hybrid CFS-RF approach. To this end, for NSL_KDD, 35 features, UNSW_NB2015, 40 features, CIC_IDS2017, and 40 features, CICDDOS2019, were obtained through the analysis and dissemination of the datasets from the planned CFS-RF.

After that, HABBA's and two other classification forms (binary and multi-class) of a confusion matrix were used to identify potential invasions. Precision, recall, DR, FNR, and FAR are distinct metrics for assessing system performance. They were implemented utilizing the following technical specifications for computer hardware and software: Sklearn Library with Python 3.8 and Colab, IntelCore i9-11900K Processor, 64GB RAM, 16GB (NVIDIA GeForce RTX 3090) CPU, 1 TB of fast SSD Storage, and Windows 11 64-bit OS.

As we intend to examine the complexity time, we used Google Colab locally to have more control over our model environment, allowing us to estimate our performance parameters, so we preferred to work offline. The Google Colab has been installed locally by installing the colab package using the pip manager on top of the Jupyter Notebook.

6.2. The Runtime and Complexity

Accuracy, complexity, overfitting, and underfitting are some of the issues the models can encounter. Some techniques for dimensionality reduction can be used, like standard-Scaler, RobustScaler, MinMaxScaler, normalization, principal component analysis (PCA), and non-negative matrix factorization (NMF). One of the potential techniques for reducing overfitting and complexity is using k-fold cross validation (CV), which is a technique that guarantees that the F1 score of the ML model does not depend on how the samples of the training set and test set were picked. In k-fold CV, the dataset is alienated into c subsets [90]. However, k-fold CV, in some cases, may fail to reduce overfitting and complexity, which can occur if the training set samples are quite small or too sparse. In such cases, we need to combine more than one method to avoid complexity and overfitting, for example, PCA and k-fold or normalization and k-fold.

Using the Big-O notation, the complexity time for the algorithm is $O(N^2)$. The algorithm's performance is proportional to the square of the size of the input elements. The algorithm uses nested loops and other operations involving quadratic time complexity. The complexity and overfitting are measured based on the training and test sets' accuracy. If the training set has high accuracy and the test set has low accuracy, the model is complex

and suffers from overfitting. As the reference [58] was the main paper, we utilized the same research methodology and reimplemented its algorithms for benchmarking with various dataset pre-processing and features. Moreover, we used the more recent dataset CICDDOS2019, and we also focused on data anomalies in 6G cellular networks. Figures 4–6 show complexity time, where complexity time can be seen as the measure of how fast or slow an algorithm will perform for the input size, which is related to the computational complexities of ML Models. In addition, Figures 4–6 show the highest and lowest values. NSL KDD, UNSW NB 2015, CIC_IDS2017, and CICDDOS2019 were used to describe the running time. The DoS class had a maximum time of 9.2 s, while the R2L class had a minimum of 1.1 s, as shown in Figure 4. Figure 4 also shows the maximum complexity time of the NSL KDD dataset.

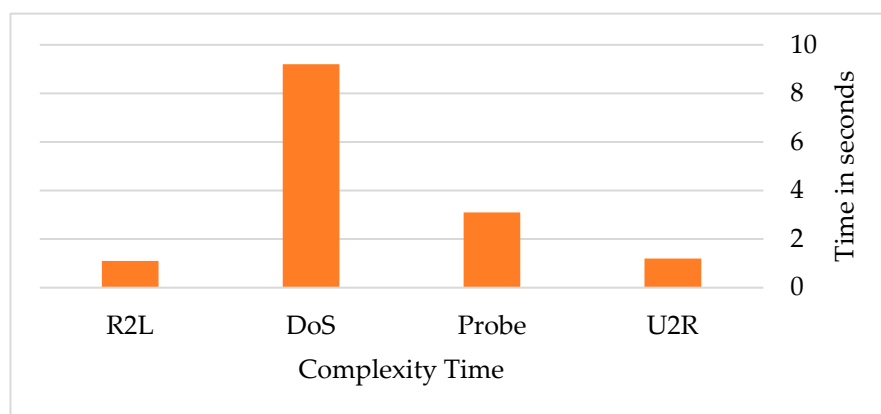


Figure 4. NSL KDD dataset complexity time.

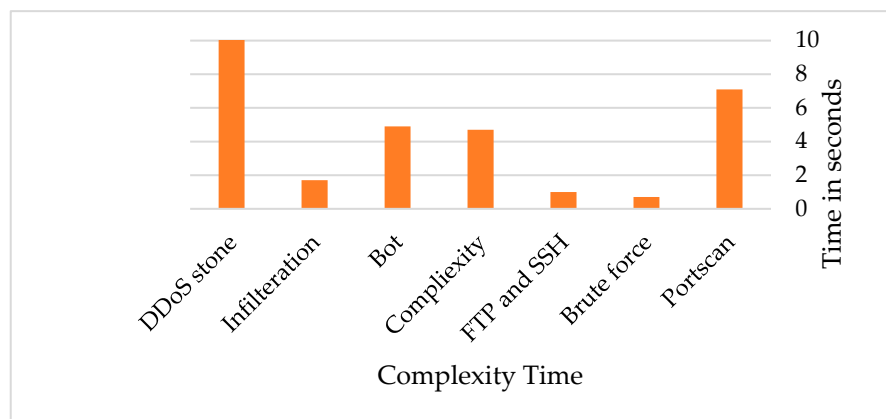


Figure 5. Dataset complexity time for UNSW NB2015.

The UNSW NB2015 dataset complexity time is shown in Figure 5. The complexity time for the CIC IDS2017 dataset, shown in Figure 6, was 6.3 s for the DDoS class and 0.8 s for the Shellcode class. According to Figure 5, the brute force class had the lowest response time at 0.7 s, and the DDoS stone class had the highest response time at 11.3 s. According to [58], the verifications show almost similar quantities for Figures 4–6. For example, in Figure 4, both results were 1.3, 1.8, 3.2, and 9.6 s for R2L, DoS, Probe, and U2R, respectively. By means of verification and validation of our work, we examined the CICDDOS2019 dataset in Figure 7 and implemented other algorithms, i.e., SVM, DT, RF, DBN, NB, and ANN, in Figures 8–10. According to the CICDDOS2019 dataset, as shown in Figure 7, the FTP and SSH classes had the lowest response time at 0.3 s, and the DDoS class had the highest response time at 6 s. As a result, the running time was proportional to the number of inputs and increased as the inputs increased.

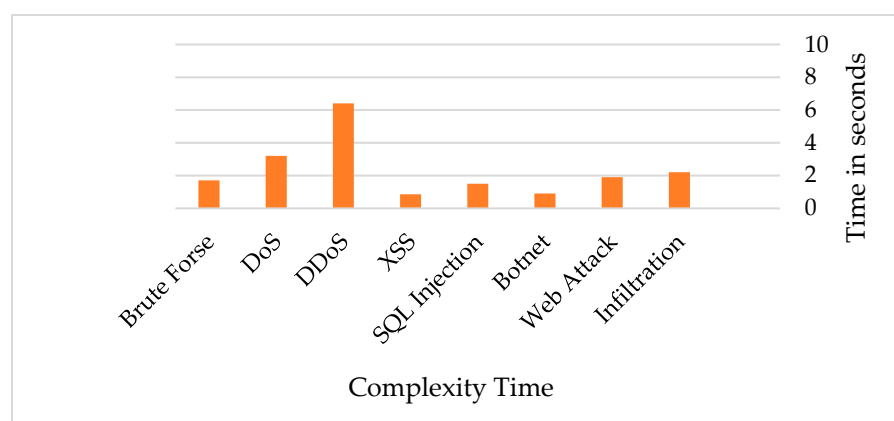


Figure 6. Dataset complexity time for CIC IDS2017.

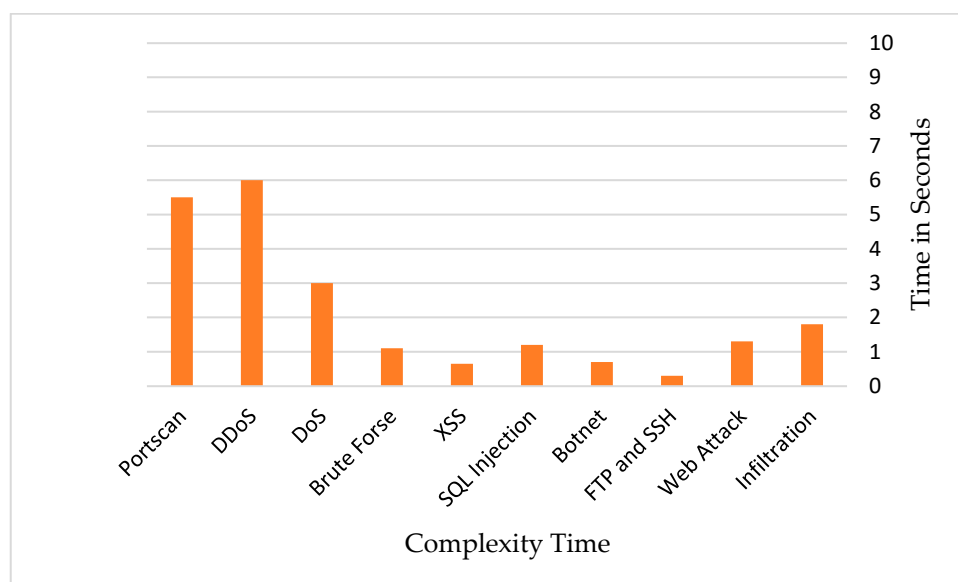


Figure 7. Dataset complexity time for CICDDOS2019.

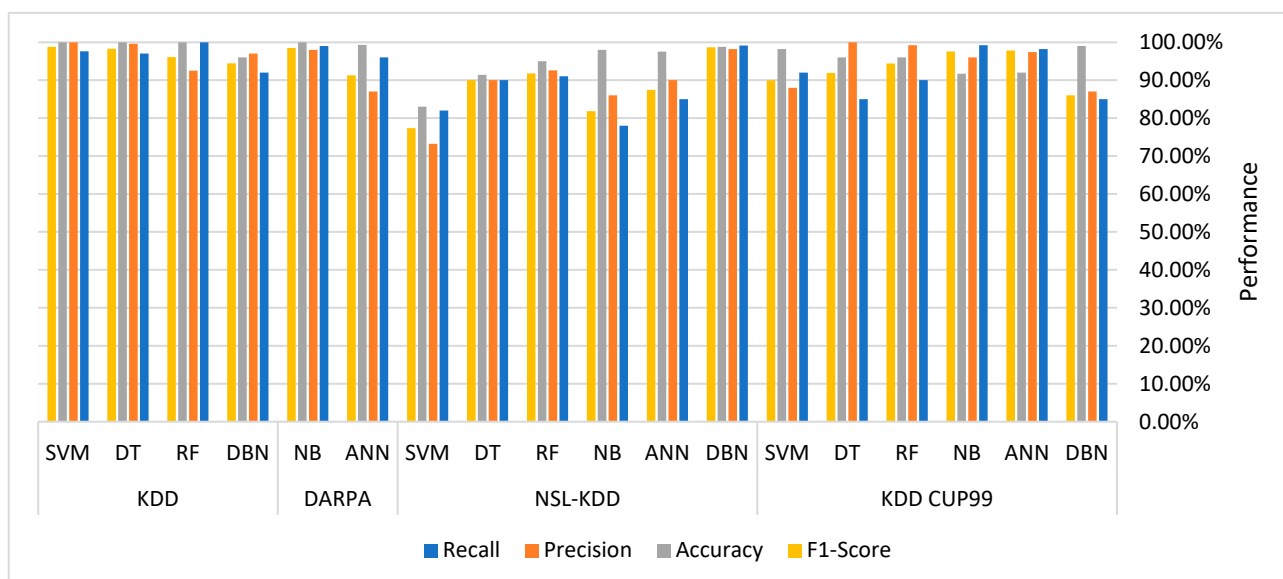


Figure 8. A comparison of machine learning algorithms for intrusion detection.

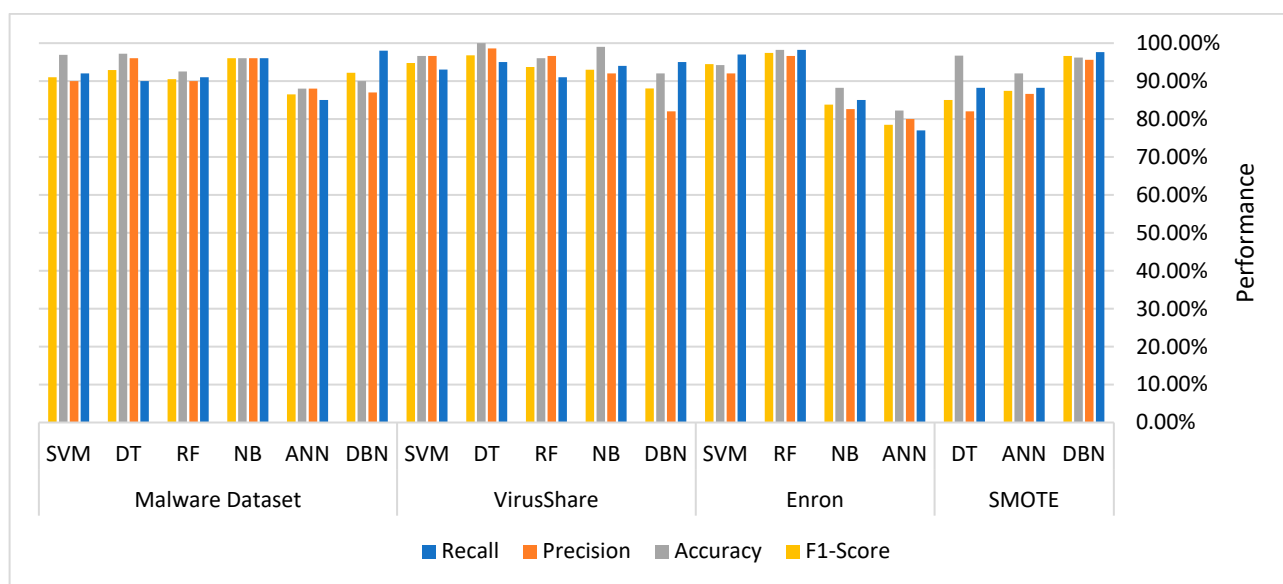


Figure 9. Malware detection comparative analysis using machine learning techniques.

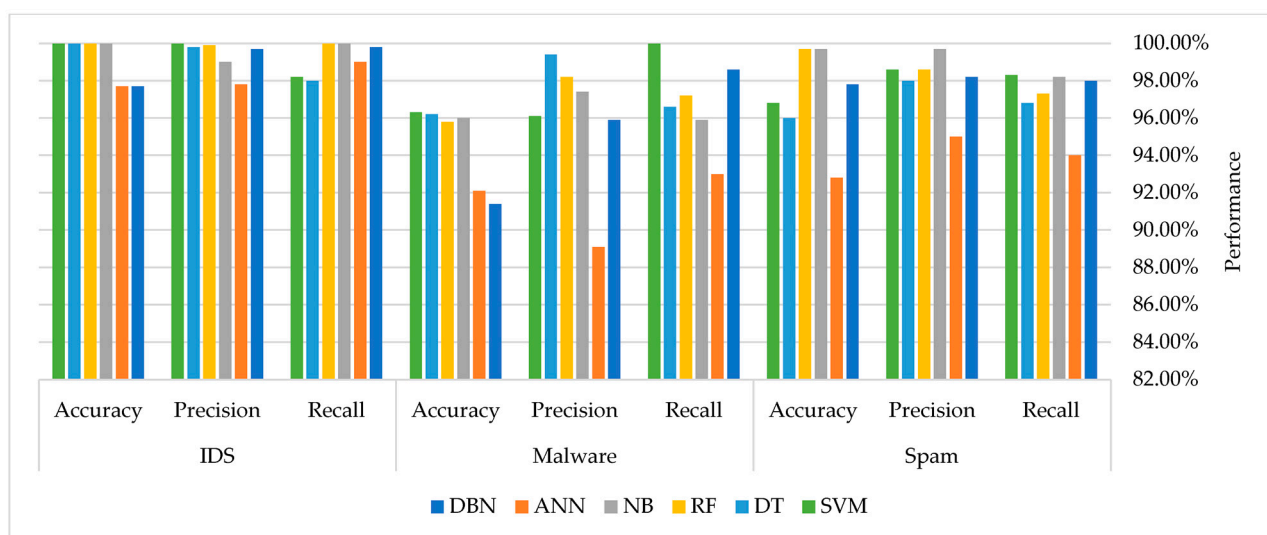


Figure 10. Machine learning techniques performance evaluation.

6.3. Restrictions

To make the system more resistant to new threats, the major goal of the study was to differentiate between regular and aberrant actions. However, the study has the following drawbacks: When using dataset assaults, the HABBAs system performs admirably but ignores additional attacks initiated by external networks (when available). It is difficult to migrate the HABBAs system to identify new threats once the training phase is over due to the results of the training component.

6.4. Evaluation of ML Models' Performance

We selected the ML methods of DBN, ANN, NB, RF, DT, and SVM since these methods are frequently used machine learning models for cybersecurity. Figure 8 compares the effectiveness of six machine-learning methods using datasets often used for intrusion detection. The values were chosen from the provided tables that, according to the dataset, represented the highest possible levels of recall, precision, accuracy, and F1-score. SVM demonstrated the greatest accuracy, reported on the NSL-KDD dataset at 83.2%. However,

the KDD dataset's performance was approximately 98.1%, whereas the F1-score was 90.99%. On practically all datasets, DBN worked excellently and demonstrated intrusion detection accuracy above 95%, and the F1-score was 92.17%.

NB and ANN outperformed other models regarding accuracy on the DARPA dataset. However, ANN provided lower precision values. DBN outperformed other models regarding accuracy, precision, and recall on the NSL-KDD dataset. On the KDD-Cup 99 dataset, SVM and DBN outperformed all other models in terms of precision. Decision trees and random forests had the best precision rates among all the models tested on the KDD dataset. In that order, the best recall rates were demonstrated by the KDD dataset with random forest, DARPA with NB, NSL with DBN, and Cup99 with KDD.

The examination of six machine learning methods' performance on widely used malware detection datasets is shown in Figure 9. There are not many benchmark datasets for malware detection. For the most part, the researchers gathered their unique datasets and used machine learning techniques to assess the models. On a tailored dataset, ML approaches frequently display exceptional F1-score, accuracy, precision, and recall values, as has been noted. The offered strategies do not exhibit comparable performance when used on different datasets.

Traditional machine learning methods, such as decision trees, fared better on several datasets. On nearly all datasets, DBN had a fantastic recall value. On the VirusShare dataset, DT and RF displayed improved precision rates. On the Enron dataset, RF demonstrated good values for recall, accuracy, and precision. ANN displayed the weakest accuracy, recall, and precision on the Enron dataset. The NB and DT compared well to other models in terms of accuracy on the VirusShare dataset.

The comparison of accuracy, precision, and recall values for detecting intrusion, spam, and malware is shown in Figure 10. No matter the dataset, the highest value was the token that six machine learning models could produce. It has been demonstrated that SVM, DT, and RF deliver the highest accuracy and precision values for intrusion detection. However, DBN and ANN had the highest recall values for intrusion detection. If intrusion detection accuracy is the highest priority, SVM, DT, NB, and RF are reimplemented as potential candidates.

Comparatively speaking, DBN and ANN were less effective than other models at detecting malware. However, ANN also demonstrated exceptional recall value. Although RF and NB have demonstrated greater accuracy in classifying spam, DBN is still advised in situations where precision and recall are crucial. For greater accuracy, it is advised to use RF and NB for classifying spam, SVM and DT for detecting malware, and DT for both tasks, considering the metrics gathered from the reviewed papers.

6.5. Analysis, Discussions and Evaluations

The categorization technique, number of FS, FS, FAR, and DR, as well as the accuracy of the reimplemented method, were all analyzed and contrasted with earlier studies. During testing, the proposal's detection accuracy was 90%, but during training, it was 99%. It produced a higher DR with a lower FAR than the single-stage technique.

Additionally, it regularly achieved the highest accuracy, DR, and FAR levels compared to past research.

The reimplemented system's accuracy, DR, FAR, the number of FS, FS, and classification methods were all studied and contrasted with prior studies. The proposal's detection accuracy was 99% during training and 90% during testing. It produced a greater FAR for the DR, which is lower than for the single-stage technique. In addition, compared to past investigations, it consistently achieved the highest levels of accuracy, DR, and FAR. This trade-off is further explained in Table 9.

6.6. Recommendations

The dataset must go through a preprocessing stage to prepare for the CFS-RF feature selection step. As part of each class in the dataset, after the CFS-RF step, it is exposed to

analysis in order to confirm and choose only the most potent influencing characteristics that affect the outcomes [86]. Finally, CFS-RF chooses the dataset's most useful feature subset, which includes 40 features for CIC IDS201, 35 for UNSW NB201, and 30 for NSL KD [87]. The classifier stage then begins. In this scenario, both SVM and RF classifiers operate sequentially as an adaboosting algorithm [88], and they compile information using the typical voting method to function concurrently as a bagging algorithm [89].

Table 9. Comparison of the reimplemented system's findings with those from other studies.

Fih	Dataset	Number of FS	Accuracy %	FAR %	DR %
[48]	NSL_KDD	N/A	81	N/A	82
[49]	NSL_KDD	16	84	4.85	83
[50]	NSL_KDD	32	86	15.01	N/A
[51]	NSL_KDD	10	79	1.01	N/A
	NSL_KDD	42		5.12	96.3
[52]	UNSW_NB2015	13	95.2	N/A	96.65
	CIC_IDS2017	8		N/A	98.94
	NSL_KDD	11	98	N/A	98.76
[53]	CIC_IDS2017	N/A	98.8	N/A	N/A
[56]	CIC_IDS2017	N/A	99	N/A	N/A
	NSL_KDD		87	0.004	99
[87]	UNSW_NB2015	30	85	N/A	N/A
	NSL_KDD,	30	99.4	0.004	99.9
[58]	UNSW_NB2015	35	99.8	0.008	99.6
	CIC_IDS2017	40	99.7	0.0012	99.4
	NSL_KDD,	30	99.5	0.0038	99.9
Our System	UNSW_NB2015	35	99.9	0.0076	99.7
	CIC_IDS2017	40	99.8	0.0009	99.5
	CICDDOS2019	40	99.95426	0.00113	99.9

7. Conclusions and Future Work

The current IDSs are still ineffective, largely due to the susceptibility of the anticipated wireless paradigms, despite having previously adopted various ML tactics to increase their performance. In the research, the CICDDOS2019 dataset is utilized and analyzed using ML models, where the CICDDOS2019 dataset includes a wider range of attacks, such as DDoS, DoS, brute force, XSS, SQL injection, botnets, web attacks, and infiltration. Susceptibility to attacks caused by predictable context (e.g., replay attack) because they rely on the limited entropy of wireless physical context to protect a shared key.

This study developed a unique IDS technique to handle imbalanced FS and EL algorithms' preferred hybrid approaches, which are based on high-dimensional traffic with less DR. Using samples from NSL KDD, UNSW NB 2015, CIC_IDS2017, and CICDDOS2019, a hybrid EL approach was used to obtain the best 30 features, 35 features, a subset of function correlation, and 40 features, respectively.

The values of the FAR for the datasets UNSW NB2015, NSL KDD, CIC_IDS2017, and CICDDOS2019 were 0.0039, 0.0076, 0.0009, and 0.00113, respectively, with an accuracy of 99.7% for all datasets. The results comparison table includes information on other parametric values. The number of records selected was inversely associated with the percentage of beginning KDDCup'99 dataset records picked for each difficulty degree categorization.

In our sample, there were 125,973 occurrences of the KDD_Train set, including 58,630 attacks and 67,343 instances of routine traffic. The majority of current low-key assaults were included in the second dataset (UNSW-NB15), which aimed to replicate current network configurations. It contained 2,540,044 records spread across 4 big-data CSV files, 45 columns (id = 1, features = 44), 175,341 training records, and 82,332 testing records. The system approach performed better than the current classification algorithms. This approach gave the IDS market a large competitive advantage over other tactics. Although

CFS-RF with ensemble HABBA algorithms has advantages, further work is still needed to strengthen the system's ability to handle potential threats from infrequent future traffic.

IDS has applied a connection record for each separately, and by putting the reimplemented NIDS on the private security server firms' systems, the authors think that looking at connections in a stream of data can be useful in identifying undetectable assaults. The provided method is an excellent and trustworthy way to identify network breaches quickly and precisely.

In regard to future work, robust machine learning models are required to handle adversarial inputs. In order to build models resilient against hostile inputs, the model should be trained in hostile scenarios. We have examined some machine learning models that use different datasets to identify a threat to 6G security. However, we recommend that a novice in this field explore this study's entire reference. Our upcoming work will examine and use more ML and DL methods to combat many other cybersecurity concerns. We will evaluate ML models in various cybersecurity domains, such as IoT, smart cities, techniques based on API calls, cellular networks, and smart grids.

In the future, we want to dig deeper into the various 6G network attacks. Future research will be required to address the crucial problem of protecting 6G, such as fuzzy logic rules. Fuzzy logic can help improve the accuracy and effectiveness of 6G security systems by considering uncertainty and imprecision in data.

Author Contributions: Conceptualization, M.M.S. and M.K.H.; methodology, M.M.S., M.K.H., R.A.M. and R.A.; software, M.M.S., M.A., R.A.S. and M.K.H.; validation, M.M.S., M.A., R.A., R.A.M. and R.A.S.; formal analysis, M.K.H., M.A. and R.A.S.; investigation, R.A.; resources, M.K.H.; data curation, M.A., R.A.S. and M.K.H.; writing—original draft preparation, R.A., M.M.S. and M.K.H.; writing—review and editing, M.A., R.A., R.A.M. and R.A.S.; visualization, M.M.S. and R.A.; supervision, M.K.H.; project administration, M.K.H., R.A.M. and R.A.; funding acquisition, M.A., R.A.M. and R.A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Princess Nourah Bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R97), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would also like to acknowledge the Deanship of Scientific Research at Taif University for funding this work.

Data Availability Statement: Not applicable.

Acknowledgments: Princess Nourah Bint Abdulrahman University Researchers Supporting Project Number (PNURSP2023R97), Princess Nourah Bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would also like to acknowledge the Deanship of Scientific Research at Taif University for funding this work.

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

ADCN	Anomaly Detection in Communication Networks
AES	Advanced Encryption Standard
AI	Artificial Intelligence
AIDS	Anomaly Intrusion Detection Systems
AKA	Authentication and Key Agreement
AO	Alternating Optimization
BCI	Brain-Computer Interfaces
CFS	Correlation FS
CFS-RF	Comparison Feature Selection and Random Forest Algorithm
CV	Cross Validation
DDoS	Distributed Denial of Service
DNN	Deep Neural Network
DoS	Denial of Service

DR	Detection Rate
EL	Ensemble Learning
FAR	False Alarm Rate
FNR	False Negative Rate
FPR	False Positive Rate
FS	Feature Selection
HABBAs	Hybrid Adaboosting Bagging Algorithms
ID	Intrusion Detection
IDS	Intrusion Detection Systems
MITM	Man-In-The-Middle
ML	Machine Learning
NIDS	Network-Based Intrusion Detection System
NMF	Non-negative Matrix Factorization
NVF	Network Function Virtualization
PCA	Principal Component Analysis
PS	Power-Splitting
R2L	Remote To Local
RF	Random Forests
SDN	Software-Defined Networking
SIDS	Signature Intrusion Detection Systems
SVM	Support Vector Machines
TCP	Transmission Control Protocol
U2R	User To Root
UDP	User Datagram Protocol
UE	User Equipment
XR	Extended Multimodal Reality

References

- Viswanathan, H.; Mogensen, P.E. Communications in the 6G Era. *IEEE Access* **2020**, *8*, 57063–57074. [\[CrossRef\]](#)
- Saeed, M.M.; Hasan, M.K.; Hassan, R.; Mokhtar, R.; Saeed, R.A.; Saeid, E.; Gupta, M. Preserving Privacy of User Identity Based on Pseudonym Variable in 5G. *Comput. Mater. Contin.* **2022**, *70*, 5551–5568. [\[CrossRef\]](#)
- Saeed, M.M.; Hasan, M.K.; Obaid, A.J.; Saeed, R.A.; Mokhtar, R.A.; Ali, E.S.; Akhtaruzzaman, M.; Amanlou, S.; Hossain, A.K.M.Z. A comprehensive review on the users' identity privacy for 5G networks. *IET Commun.* **2022**, *16*, 384–399. [\[CrossRef\]](#)
- Chen, Y.; Weng, Q.; Tang, L.; Wang, L.; Xing, H.; Liu, Q. Developing an intelligent cloud attention network to support global urban green spaces mapping. *ISPRS J. Photogramm. Remote. Sens.* **2023**, *198*, 197–209. [\[CrossRef\]](#)
- Banafaa, M.; Shayea, I.; Din, J.; Azmi, M.H.; Alashbi, A.; Daradkeh, Y.I.; Alhammadi, A. 6G Mobile Communication Technology: Requirements, Targets, Applications, Challenges, Advantages, and Opportunities. *Alex. Eng. J.* **2023**, *64*, 245–274. [\[CrossRef\]](#)
- Saad, W.; Bennis, M.; Chen, M. A Vision of 6G Wireless Systems: Applications, Trends, Technologies, and Open Research Problems. *IEEE Netw.* **2019**, *34*, 134–142. [\[CrossRef\]](#)
- De Alwis, C.; Kalla, A.; Pham, Q.V.; Kumar, P.; Dev, K.; Hwang, W.J.; Liyanage, M. Survey on 6G frontiers: Trends, applications, requirements, technologies and future research. *IEEE Open J. Commun. Soc.* **2021**, *2*, 836–886.
- Giordani, M.; Polese, M.; Mezzavilla, M.; Rangan, S.; Zorzi, M. Toward 6G Networks: Use Cases and Technologies. *IEEE Commun. Mag.* **2020**, *58*, 55–61. [\[CrossRef\]](#)
- Ziegler, V.; Viswanathan, H.; Flinck, H.; Hoffmann, M.; Raisanen, V.; Hatonen, K. 6G Architecture to Connect the Worlds. *IEEE Access* **2020**, *8*, 173508–173520. [\[CrossRef\]](#)
- Saeed, M.M.; Saeed, R.A.; Mokhtar, R.A.; Alhumyani, H.; Ali, E.S. A Novel Variable Pseudonym Scheme for Preserving Privacy User Location in 5G Networks. *Secur. Commun. Netw.* **2022**, 7487600. [\[CrossRef\]](#)
- Saeed, M.M.; Saeed, R.A.; Saeid, E. Survey of privacy of user identity in 5G: Challenges and proposed solutions. *Inf. Technol. Netw.* **2019**, *7*, 2312–4989.
- Saeed, R.A.; Saeed, M.M.; Mokhtar, R.A.; Alhumyani, H.; Abdel-Khalek, S. Pseudonym Mutable Based Privacy for 5G User Identity. *Comput. Syst. Sci. Eng.* **2021**, *39*, 1–14. [\[CrossRef\]](#)
- Saeed, M.M.; Saeed, R.A.; Azim, M.A.; Ali, E.S.; Mokhtar, R.A.; Khalifa, O. Green Machine Learning Approach for QoS Improvement in Cellular Communications. In Proceedings of the 2022 IEEE 2nd International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA), Sabratha, Libya, 23–25 May 2022; pp. 523–528.
- Saeed, M.M.; Ali, E.S.; Saeed, R.A. Data-Driven Techniques and Security Issues. In *Wireless Networks. In Data-Driven Intelligence in Wireless Networks: Concepts, Solutions, and Applications*, 1st ed.; Afzal, M.A., Ateeq, M., Kim, S.W., Eds.; CRC Press: Boca Raton, FL, USA, 2023; pp. 107–154.

15. Liang, W.; Xiao, L.; Zhang, K.; Tang, M.; He, D.; Li, K.C. Data fusion approach for collaborative anomaly intrusion detection in blockchainbased systems. *IEEE Internet Things J.* **2022**, *9*, 14741–14751.
16. Rajagopal, S.; Kundapur, P.P.; Hareesha, K.S. A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets. *Secur. Commun. Netw.* **2020**, 4586875. [\[CrossRef\]](#)
17. Mokhtari, S.; Abbaspour, A.; Yen, K.K.; Sargolzaei, A. A Machine Learning Approach for Anomaly Detection in Industrial Control Systems Based on Measurement Data. *Electronics* **2021**, *10*, 407. [\[CrossRef\]](#)
18. Safaldin, M.; Otair, M.; Abualigah, L. Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *12*, 1559–1576. [\[CrossRef\]](#)
19. Moon, S.-H.; Kim, Y.-H. An improved forecast of precipitation type using correlation-based feature selection and multinomial logistic regression. *Atmos. Res.* **2020**, *240*, 104928. [\[CrossRef\]](#)
20. Mohamad, M.; Selamat, A.; Krejcar, O.; Crespo, R.G.; Herrera-Viedma, E.; Fujita, H. Enhancing Big Data Feature Selection Using a Hybrid Correlation-Based Feature Selection. *Electronics* **2021**, *10*, 2984. [\[CrossRef\]](#)
21. Loey, M.; Manogaran, G.; Taha, M.H.N.; Khalifa, N.E.M. A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. *Measurement* **2020**, *167*, 108288. [\[CrossRef\]](#)
22. Panchiwala, S.; Shah, M. Information, and Management. A comprehensive study on critical security issues and challenges of the IoT world. *J. Data Inf. Manag.* **2020**, *2*, 257–278.
23. El Mrabet, Z.; Kaabouch, N.; El Ghazi, H.; El Ghazi, H. Cyber-security in smart grid: Survey and challenges. *Comput. Electr. Eng.* **2018**, *67*, 469–482. [\[CrossRef\]](#)
24. Esiner, E.; Mashima, D.; Chen, B.; Kalbarczyk, Z.; Nicol, D. F-Pro: A Fast and Flexible Provenance-Aware Message Authentication Scheme for Smart Grid. In Proceedings of the 2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), Beijing, China, 21–23 October 2019; pp. 1–7. [\[CrossRef\]](#)
25. Fraga-Lamas, P.; Lopes, S.I.; Fernández-Caramés, T.M. Green IoT and edge AI as key technological enablers for a sus-tainable digital transition towards a smart circular economy: An industry 5.0 use case. *Sensors* **2021**, *21*, 5745.
26. Leng, J.; Sha, W.; Wang, B.; Zheng, P.; Zhuang, C.; Liu, Q.; Wang, L. Industry 5.0: Prospect and retrospect. *J. Manuf. Syst.* **2022**, *65*, 279–295.
27. Fatima, Z.; Tanveer, M.H.; Waseemullah; Zardari, S.; Naz, L.F.; Khadim, H.; Ahmed, N.; Tahir, M. Production Plant and Warehouse Automation with IoT and Industry 5.0. *Appl. Sci.* **2022**, *12*, 2053. [\[CrossRef\]](#)
28. Hasan, M.K.; Saeed, R.A.; Alsuhibany, S.A.; Abdel-Khalek, S. An Empirical Model to Predict the Diabetic Positive Using Stacked Ensemble Approach. *Front. Public Health* **2022**, *9*, 792124. [\[CrossRef\]](#)
29. Brotsis, S.; Kolokotronis, N.; Limniotis, K.; Bendiab, G.; Shiales, S. On the Security and Privacy of Hyperledger Fabric: Challenges and Open Issues. In Proceedings of the 2020 IEEE World Congress on Services (SERVICES), Beijing, China, 18–23 October 2020; pp. 197–204. [\[CrossRef\]](#)
30. Bansod, S.; Ragha, L. Challenges in making blockchain privacy compliant for the digital world: Some measures. *Sādhanā* **2022**, *47*, 168. [\[CrossRef\]](#)
31. Ylianttila, M.; Kantola, R.; Gurtov, A.; Mucchi, L.; Oppermann, I.; Yan, Z.; Röning, J. 6G white paper: Research challenges for trust, security and privacy. *arXiv* **2020**, arXiv:2004.11665.
32. Fortino, G.; Guerrieri, A.; Pace, P.; Savaglio, C.; Spezzano, G. IoT Platforms and Security: An Analysis of the Leading Industrial/Commercial Solutions. *Sensors* **2022**, *22*, 2196. [\[CrossRef\]](#)
33. Sandeepa, C.; Siniarski, B.; Kourtellis, N.; Wang, S.; Liyanage, M. A survey on privacy for B5G/6G: New privacy challenges, and research directions. *J. Ind. Inf. Integr.* **2022**, 100405.
34. Khan, A.A.; Laghari, A.A.; Shaikh, A.A.; Dootio, M.A.; Estrela, V.V.; Lopes, R.T. A blockchain security module for brain-computer interface (BCI) with Multimedia Life Cycle Framework (MLCF). *Neurosci. Inform.* **2021**, *2*, 100030. [\[CrossRef\]](#)
35. Bernal, S.L.; Celdrán, A.H.; Pérez, G.M.; Barros, M.T.; Balasubramaniam, S. Security in brain-computer interfaces: State-of-the-art, opportunities, and future challenges. *ACM Comput. Surv.* **2021**, *54*, 1–35.
36. Ajrawi, S.; Rao, R.; Sarkar, M. Cybersecurity in Brain-Computer Interfaces: RFID-based design-theoretical framework. *Inform. Med. Unlocked* **2020**, *22*, 100489. [\[CrossRef\]](#)
37. Wei, T.; Feng, W.; Chen, Y.; Wang, C.-X.; Ge, N.; Lu, J. Hybrid Satellite-Terrestrial Communication Networks for the Maritime Internet of Things: Key Technologies, Opportunities, and Challenges. *IEEE Internet Things J.* **2021**, *8*, 8910–8934. [\[CrossRef\]](#)
38. Wang, M.; Zhu, T.; Zhang, T.; Zhang, J.; Yu, S.; Zhou, W. Security and privacy in 6G networks: New areas and new challenges. *Digit. Commun. Netw.* **2020**, *6*, 281–291. [\[CrossRef\]](#)
39. Jiang, H.; Mukherjee, M.; Zhou, J.; Lloret, J. Channel modeling and characteristics for 6G wireless communications. *IEEE Netw.* **2020**, *35*, 296–303.
40. Promwongsa, N.; Ebrahimzadeh, A.; Naboulsi, D.; Kianpisheh, S.; Belqasmi, F.; Glitho, R.; Crespi, N.; Alfandi, O. A Comprehensive Survey of the Tactile Internet: State-of-the-Art and Research Directions. *IEEE Commun. Surv. Tutor.* **2020**, *23*, 472–523. [\[CrossRef\]](#)
41. Abdelsamad, S.E.; Abdelteef, M.A.; Elsheikh, O.Y.; Ali, Y.A.; Elsonni, T.; Abdelhaq, M.; Alsaqour, R.; Saeed, R.A. Vision-Based Support for the Detection and Recognition of Drones with Small Radar Cross Sections. *Electronics* **2023**, *12*, 2235. [\[CrossRef\]](#)
42. Khan, A.; Li, J.P.; Hasan, M.K.; Varish, N.; Mansor, Z.; Islam, S.; Saeed, R.A.; Alshammari, M.; Alhumyani, H. PackerRobo: Model-based robot vision self supervised learning in CART. *Alex. Eng. J.* **2022**, *61*, 12549–12566. [\[CrossRef\]](#)

43. Mahmood, M.R.; Matin, M.A.; Sarigiannidis, P.; Goudos, S.K. comprehensive review on artificial intelligence/machine learning algorithms for em-powering the future IoT toward 6G era. *IEEE Access* **2022**, *10*, 87535–87562.
44. Thien, H.T.; Tuan, P.-V.; Koo, I. A Secure-Transmission Maximization Scheme for SWIPT Systems Assisted by an Intelligent Reflecting Surface and Deep Learning. *IEEE Access* **2022**, *10*, 31851–31867. [\[CrossRef\]](#)
45. Uysal, D.T.; Yoo, P.D.; Taha, K. Data-driven malware detection for 6G networks: A survey from the perspective of continuous learning and explainability via visualisation. *IEEE Open J. Veh. Technol.* **2022**, *4*, 61–71.
46. Ozpoyraz, B.; Dogukan, A.T.; Gevez, Y.; Altun, U.; Basar, E. Deep Learning-Aided 6G Wireless Networks: A Comprehensive Survey of Revolutionary PHY Architectures. *arXiv* **2022**, arXiv:2201.03866.
47. Letaief, K.B.; Shi, Y.; Lu, J.; Lu, J. Edge artificial intelligence for 6G: Vision, enabling technologies, and applications. *IEEE J. Sel. Areas Commun.* **2021**, *40*, 5–36.
48. Johnson, J.M.; Yadav, A. Fault Location Estimation in HVDC Transmission Line Using ANN. In Proceedings of the First International Conference on Information and Communication Technology for Intelligent Systems: Volume 1 (Smart Innovation, Systems and Technologies), Ahmedabad, India, 28–29 November 2015; pp. 205–211. [\[CrossRef\]](#)
49. Alatabani, L.E.; Ali, E.S.; Mokhtar, R.A.; Saeed, R.A.; Alhumyani, H.; Hasan, M.K. Deep and Reinforcement Learning Technologies on Internet of Vehicle (IoV) Applications: Current Issues and Future Trends. *J. Adv. Transp.* **2022**, *2022*, 1947886. [\[CrossRef\]](#)
50. Pajouh, H.H.; Dastghaibafard, G.; Hashemi, S. Two-tier network anomaly detection model: A machine learning approach. *J. Intell. Inf. Syst.* **2015**, *48*, 61–74. [\[CrossRef\]](#)
51. Kanakarajan, N.K.; Muniasamy, K. Improving the Accuracy of Intrusion Detection using Gar-Forest with Feature Selection. In Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA) 2015; Springer: Cham, Switzerland, 2016; pp. 539–547.
52. Khalifa, O.O.; Wajdi, M.H.; Saeed, R.A.; Hashim, A.H.A.; Ahmed, M.Z.; Ali, E.S. Vehicle Detection for Vision-Based Intelligent Transportation Systems Using Convolutional Neural Network Algorithm. *J. Adv. Transp.* **2022**, *2022*, 9189600. [\[CrossRef\]](#)
53. Jaw, E.; Wang, X. Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach. *Symmetry* **2021**, *13*, 1764. [\[CrossRef\]](#)
54. Gupta, N.; Jindal, V.; Bedi, P. CSE-IDS: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems. *Comput. Secur.* **2021**, *112*, 102499. [\[CrossRef\]](#)
55. Mighan, S.N.; Kahani, M. A novel scalable intrusion detection system based on deep learning. *Int. J. Inf. Secur.* **2021**, *20*, 387–403.
56. De Souza, C.A.; Westphall, C.B.; Machado, R.B.; Sobral, J.B.M.; dos Santos Vieira, G. Hybrid approach to intrusion detection in fog-based IoT environments. *Comput. Netw.* **2020**, *180*, 107417.
57. Mhawi, D.N.; Aldallal, A.; Hassan, S. Advanced feature-selection-based hybrid ensemble learning algorithms for network intrusion detection systems. *Symmetry* **2022**, *14*, 1461.
58. Oleiwi, H.W.; Mhawi, D.N.; Al-Raweshidy, H. MLTs-ADCNs: Machine Learning Techniques for Anomaly Detection in Communication Networks. *IEEE Access* **2022**, *10*, 91006–91017. [\[CrossRef\]](#)
59. Gawali, V.S.; Nihar, M.R. Anomaly detection system in 5G networks via deep learning model. *Int. J. Wirel. Mob. Comput.* **2023**, *24*, 287–302.
60. Koursioupas, N.; Magoula, L.; Barmounakis, S.; Stavarakakis, I. Network Traffic Anomaly Prediction for Beyond 5G Networks. In Proceedings of the 2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Kyoto, Japan, 12–15 September 2022.
61. Lazar, V.; Buzura, S.; Iancu, B.; Dadarlat, V. Anomaly Detection in Software Defined Wireless Sensor Networks Using Recurrent Neural Networks. In Proceedings of the 2021 IEEE 17th International Conference on Intelligent Computer Communication and Processing (ICCP), Cluj-Napoca, Romania, 28–30 October 2021.
62. Mhawi, D.N.; Hashem, S.H. Proposed Hybrid Correlation Feature Selection Forest Panalized Attribute Approach to advance IDSs. *Symmetry* **2021**, *7*, 15. [\[CrossRef\]](#)
63. Saeed, R.A.; Omri, M.; Abdel-Khalek, S.; Ali, E.S.; Alotaibi, M.F. Optimal path planning for drones based on swarm intelligence algorithm. *Neural Comput. Appl.* **2022**, *34*, 10133–10155. [\[CrossRef\]](#)
64. Kulariya, M.; Saraf, P.; Ranjan, R.; Gupta, G.P. Performance analysis of network intrusion detection schemes using Apache Spark. In Proceedings of the 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, India, 6–8 April 2016; pp. 1973–1977.
65. Bakkiam David, D.; Al-Turjman, F. Synonym-based multi-keyword ranked search with secure k-NN in 6G network. *IET Netw.* **2022**, 1–12.
66. Fischer, E.A. *Creating a National Framework for Cybersecurity: An Analysis of Issues and Options*; Nova Science Publishers: Hauppauge, NY, USA, 2009.
67. Dar, K.S.; Javed, I.; Ammar, S.A.; Abbas, S.K.; Asghar, S.; Bakar, M.A.; Shaukat, U. A survey-data privacy through different methods. *J. Netw. Commun. Emerg. Technol.* **2015**, *5*, 1–7.
68. Purkait, M. Phishing counter measures and their effectiveness—literature review. *Inf. Manag. Comput. Secur.* **2012**, *20*, 382–420.
69. Shelly, G.B.; Vermaat, M.E. *Discovering Computers-Fundamentals 2011 Edition*; Course Technology Press: Boston, MA, USA, 2010.
70. Siriwardhana, Y.; Porambage, P.; Liyanage, M.; Ylianttila, M. AI and 6G security: Opportunities and Challenges. In Proceedings of the 2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Porto, Portugal, 8–11 June 2021.

71. Giordani, M.; Zorzi, N. Non-terrestrial networks in the 6G era: Challenges and opportunities. *IEEE Netw.* **2020**, *35*, 244–251.
72. Ali, E.S.; Hasan, M.K.; Hassan, R.; Saeed, R.A.; Hassan, M.B.; Islam, S.; Nafi, N.S.; Bevinakoppa, S. Machine Learning Technologies for Secure Vehicular Communication in Internet of Vehicles: Recent Advances and Applications. *Wiley-Hindawi J. Secur. Commun. Netw.* **2021**, 8868355. [[CrossRef](#)]
73. Plastiras, G.; Terzi, M.; Kyrkou, C.; Theodoridis, T. Edge intelligence: Challenges and opportunities of near-sensor machine learning applications. In Proceedings of the 2018 IEEE 29th International Conference on Application-Specific Systems, Architectures and Processors (Asap), Milan, Italy, 10–12 July 2018; pp. 1–7.
74. Peng, H.; Wang, Z.; Han, S.; Jiang, Y. Physical layer security for MISO NOMA VLC system under eaves-dropper collusion. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6249–6254.
75. Chih-Lin, I. AI as an Essential Element of a Green 6G. *IEEE Trans. Green Commun. Netw.* **2021**, *5*, 1–3.
76. Ahmed, E.S.A.; Mohammed, Z.T.; Hassan, M.B.; Saeed, R.A. Algorithms Optimization for Intelligent IoV Applications. In *Handbook of Research on Innovations and Applications of AI, IoT, and Cognitive Technologies*; Zhao, J., Kumar, V.V., Eds.; IGI Global: Hershey, PA, USA, 2021; pp. 1–25. [[CrossRef](#)]
77. Zhang, S.; Zhu, D. Towards artificial intelligence enabled 6G: State of the art, challenges, and opportunities. *Comput. Netw.* **2020**, *183*, 107556.
78. Khalifa, O.O.; Roubleh, A.; Esgiar, A.; Abdelhaq, M.; Alsaqour, R.; Abdalla, A.; Ali, E.S.; Saeed, R. An IoT-Platform-Based Deep Learning System for Human Behavior Recognition in Smart City Monitoring Using the Berkeley MHAD Datasets. *Systems* **2022**, *10*, 177. [[CrossRef](#)]
79. Qiao, X.; Huang, Y.; Dustdar, S.; Chen, J. 6G vision: An AI-driven decentralized network and service architecture. *IEEE Internet Comput.* **2020**, *24*, 33–40.
80. Zhang, Z.; Xiao, Y.; Ma, Z.; Xiao, M.; Ding, Z.; Lei, X.; Fan, P. 6G wireless networks: Vision, requirements, architecture, and key technologies. *IEEE Veh. Technol. Mag.* **2019**, *14*, 28–41.
81. Sattiraju, R.; Weinand, A.; Schotten, H.D. AI-assisted PHY technologies for 6G and beyond wireless networks. *arXiv* **2019**, arXiv:1908.09523.
82. Hong, T.; Liu, C.; Kadoch, M. Machine learning based antenna design for physical layer security in ambient backscatter communications. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 4870656.
83. Elfatih, N.M.; Hasan, M.K.; Kamal, Z.; Gupta, D.; Saeed, R.A.; Ali, E.S.; Hosain, S. Internet of vehicle's resource management in 5G networks using AI technologies: Current status and trends. *IET Commun.* **2021**, *16*, 400–420. [[CrossRef](#)]
84. Nawaz, S.J.; Sharma, S.K.; Wyne, S.; Patwary, M.N. Asaduzzaman Quantum Machine Learning for 6G Communication Networks: State-of-the-Art and Vision for the Future. *IEEE Access* **2019**, *7*, 46317–46350. [[CrossRef](#)]
85. Zhou, Z.; Liao, H.; Gu, B.; Huq, K.M.S.; Mumtaz, S.; Rodriguez, J. Robust Mobile Crowd Sensing: When Deep Learning Meets Edge Computing. *IEEE Netw.* **2018**, *32*, 54–60. [[CrossRef](#)]
86. Mansour, R.F.; Alfar, N.M.; Abdel-Khalek, S.; Abdelhaq, M.; Saeed, R.A.; Alsaqour, R. Optimal deep learning based fusion model for biomedical image classification. *Expert Syst.* **2021**, *39*, e12764. [[CrossRef](#)]
87. Tomkos, I.; Klonidis, D.; Pikasis, E.; Theodoridis, S. Toward the 6G Network Era: Opportunities and Challenges. *IT Prof.* **2020**, *22*, 34–38. [[CrossRef](#)]
88. Gadal, S.; Mokhtar, R.; Abdelhaq, M.; Alsaqour, R.; Ali, E.S.; Saeed, R. Machine Learning-Based Anomaly Detection Using K-Mean Array and Sequential Minimal Optimization. *Electronics* **2022**, *11*, 2158. [[CrossRef](#)]
89. Alqurashi, F.A.; Alsolami, F.; Abdel-Khalek, S.; Ali, E.S.; Saeed, R.A. Machine learning techniques in internet of UAVs for smart cities applications. *J. Intell. Fuzzy Syst.* **2022**, *42*, 3203–3226. [[CrossRef](#)]
90. Abdou, A.; Van Oorschot, P.C.; Wan, T. Comparative analysis of control plane security of SDN and conventional networks. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 3542–3559.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.