data

MDPI

Article

Cloud-Based Smart Contract Analysis in FinTech Using IoT-Integrated Federated Learning in Intrusion Detection

Venkatagurunatham Naidu Kollu [1], Vijayaraj Janarthanan [2], Muthulakshmi Karupusamy [3] and Manikandan Ramachandran [4,*]

[1] Department of Computer Science and Engineering, Dr.M.G.R. Educational and Research Institute, Chennai 600095, India
[2] Department of Artificial Intelligence and Data Science, Easwari Engineering College, Ramapuram, Chennai 600089, India
[3] Department of Information Technology, PanimalarEngineering College, Poonamallee, Chennai 600123, India
[4] School of Computing, SASTRA Deemed University, Thanjavur 613401, India
* Correspondence: manikandan75@core.sastra.edu

Abstract: Data sharing is proposed because the issue of data islands hinders advancement of artificial intelligence technology in the 5G era. Sharing high-quality data has a direct impact on how well machine-learning models work, but there will always be misuse and leakage of data. The field of financial technology, or FinTech, has received a lot of attention and is growing quickly. This field has seen the introduction of new terms as a result of its ongoing expansion. One example of such terminology is "FinTech". This term is used to describe a variety of procedures utilized frequently in the financial technology industry. This study aims to create a cloud-based intrusion detection system based on IoT federated learning architecture as well as smart contract analysis. This study proposes a novel method for detecting intrusions using a cyber-threat federated graphical authentication system and cloud-based smart contracts in FinTech data. Users are required to create a route on a world map as their credentials under this scheme. We had 120 people participate in the evaluation, 60 of whom had a background in finance or FinTech. The simulation was then carried out in Python using a variety of FinTech cyber-attack datasets for accuracy, precision, recall, F-measure, AUC (Area under the ROC Curve), trust value, scalability, and integrity. The proposed technique attained accuracy of 95%, precision of 85%, RMSE of 59%, recall of 68%, F-measure of 83%, AUC of 79%, trust value of 65%, scalability of 91%, and integrity of 83%.

Keywords: intrusion detection system; FinTech; IoT federated learning architecture; smart contract analysis; cloud computing

Citation: Kollu, V.N.; Janarthanan, V.; Karupusamy, M.; Ramachandran, M. Cloud-Based Smart Contract Analysis in FinTech Using IoT-Integrated Federated Learning in Intrusion Detection. *Data* **2023**, *8*, 83. https://doi.org/10.3390/data8050083

Academic Editor: Keke Chen

Received: 13 March 2023
Revised: 24 April 2023
Accepted: 24 April 2023
Published: 29 April 2023

1. Introduction

It has been stated that the fields of ML (machine learning) as well as data science play a crucial role in science. As a result, the need for privacy remains a fundamental value for the majority of people. The idea of collecting data is becoming more popular all over the world. It has also been demonstrated that social media platforms can improve services by collecting user data [1]. The amount of data used has significantly increased, resulting in an increase in private data. As a result, privacy-related concepts have attracted the attention of consumers and policymakers. General Data Protection Regulation (GDPR) is among the moves made because of information security across globe. The move has been linked to the need to promote the necessary kinds of developments and ensure that significant accomplishments in the field of protected data are made. In 2017, Google introduced the concept of FL [2]. The idea empowered information researchers to share their factual models in dissecting information. However, security was an essential requirement for data sharing. As a result, federated learning proved to be a successful strategy for facilitating privacy

for data analysts all over the world. Blockchain technology has begun to be recognized by financial services as having the potential to revolutionize areas such as increasing revenue, enhancing the end-user experience, and the delivery process [3]. Similarly to every other technology-focused sector, the financial technology industry is currently developing. There are a great deal of new monetary applications coming out constantly, and every one offers better and more innovative ways of taking care of installments and cycling them. It is anticipated that financial blockchain industry will have reached a value of 36.04 billion dollars by the end of 2028 [4].

An emerging financial technology known as "Decentralized Finance," or "DeFi," is based on the blockchain and aims to limit banks' control over money and financial services. Digital ledgers will also have a big impact on how we obtain, send, store, and manage our money over many decades. Cybercriminals can target almost any business, but they typically select their targets based on which ones will bring in the most revenue or have the greatest impact. Because they satisfy the aforementioned requirements, banks and other similar financial services are frequently targeted by hackers. Their continuous endeavors to carefully change as well as the difficult strategic situation that is accelerating the utilization of hybridwork areas all make it more straightforward for cybercriminals to get information and sell it. Cyber threat actors are disproportionately concentrating on the banking sector as a direct result of this [5]. This is the basis for complex joint model training, which makes these benefits possible [6]. For training this proposed model, we obtained datasets from https://www.unb.ca/cic/datasets/nsl.html, https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data, https://www.unb.ca/cic/datasets/ids.html (all accessed on 9 January 2023) with data license from Canadian Institute for Cybersecurity.

Research Gap: Ongoing advancements in deep-learning procedures have delivered critical enhancements in well-established simulated intelligence occupations, such as medication revelation, quality examination, and discourse and picture acknowledgment. According to McMahan et al., despite the numerous benefits of deep learning, the same training dataset that made it so reliable also raises serious privacy concerns. Federated Deep Learning is a mobile device-specific distributed deep-learning paradigm (FDL). In FDL, distributed training involves a number of parties, and a parameter server keeps track of a developing deep-learning model. This effectively combines distributed computation with deep learning.

Research Goal: The cyber threat federated graphical authentication system is used to develop the proposed model for cloud-based smart contracts in FinTech data and their intrusion detection. Utilizing artificial intelligence and blockchain technology has numerous benefits. Simply put, blockchain's inherent security features can be strengthened by utilizing machine learning's analytical capabilities. The capacity to handle enormous measures of information safely and really in the monetary administrations industry might be very significant to foundations and end clients.

Research Questions: The industrial Internet of Things (IIoT) has radically changed over the past few years as a result of the swift advancement of wireless transmission and processing. IoT networks have seen the emergence of a wide variety of cutting-edge portable devices, including smart phones, smart watches, and smart apps. They have been heavily utilized by a variety of businesses, including live gaming, smart manufacturing, navigational systems, smart cities, and smart healthcare. Because of their rapid adoption, there are still a number of significant problems with IoT network design. One of the main issues is finding effective and adaptable control for IoT systems that might help with energy conservation, increase the number of applications, and make future development easier. Other key barriers are the cognitive load, time efficiency, and ongoing need for computer resources in IoT networks. Additionally, they ensure confidentiality and protection against unauthorized access. Due to the rapid advancement of personal awareness and digital technology, people are starting to take personal data security even more seriously. Distributed learning techniques are necessary for devices to cooperate to produce a single learning approach with local training. Federated learning (FL) is a decentralized machine-learning

(ML) platform. In contrast to centralized learning frameworks, data created on an end device do not leave the device, hence the FL framework naturally encourages secrecy and privacy. The distributed learning model is trained on the device itself using the data from the participating devices. A client device (such a neighbourhood Wi-Fi switch) and a cloud server only provide the modified settings. The following are some benefits of using FL in remote IoT networks: (i) local ML system settings can save power and use less wireless bandwidth than exchanging a lot of training data; (ii) by locally calibrating an ML model's parameters, a significant reduction in transmission delay can be achieved; and (iii) FL can help with information protection because the majority of the local learning model factors are sent, and the preparation information is stored locally.

2. Literature Review

Every financial institution wants to develop a safe business operating system to increase its level of security without having to install a lot of hardware resources [7]. When electronic transactions and networking methods first emerged, the financial industry's cyber problem was initially viewed as a business issue [8]. A number of surveys have highlighted the importance of developing solid IT security strategies [9]. The use of various security measures to safeguard sensitive data is the subject of one of the surveys about business operations.

Existing FinTech Intrusion Detection System

A recent study by [10] investigates whether or not the financial institution's records are protected against current attacks. In addition, the study measured the potential effects of security risks on business scope, sales, performance, and markets. In addition, Ref. [11] has demonstrated how much IT transparency can affect the trustworthiness of financial counseling encounters. Overall, concerns about business operations stem from an unidentified technicality and the concealment of the implementation process and IT strategy formulation [12]. One application of FinTech for enhancing business operations is the use of intelligent agents to predict and monitor financial risks [13]. In general, the implementation process, IT strategy formulation, and unknown technical details are the root causes of most business operations-related concerns. Some hidden technical components arise as a result of business operations concerns. The classification of cyber incidents and masked technical complexity have prompted the majority of FSI cyber concerns. In addition, the FinTech industry has largely accepted cloud computing as a web-based service model [14]. For instance, Bank of America (BoA) recently announced that Microsoft and financial institutions are collaborating on the development of blockchain technologies to improve financial transitions [15]. Cloud-based solutions boost system performance by connecting financial companies and their intended markets in close proximity [16]. However, because workload is outsourced, this company also introduced new threats and attacks. The most pressing issues with cloud-based solutions are: a) an absence of information controls in mists that make veil intricacy issues for FSIs [17] on the grounds that private mists become standard in FinTech. Federated ML was the focus of work [18] and author [19] presentations of FL's current advanced and unsolved issues. Numerous researchers focused on FL's issues, and work [20] conducted a survey of FL system components regarding privacy as well as security. In mobile-edge computing (MEC), author [21] provided an overview of FL chain's concepts and opportunities. Author [22] provided a comprehensive analysis of FL privacy and security that has the potential to help bridge the gap between the current state of federated AI (FAI)and the future. Work [23] discussed the privacy issue and preservation measures surrounding the integration of BT and FL for IoT. The summary based on the existing technique is added in Table 1.

Table 1. Summary based on existing FL-based FinTech intrusion detection.

Applicable Domain	Objective	Contribution	Limitation
Blockchain and AI	A study on AI-related blockchain applications	The literature reviews on new blockchain platforms, applications, and protocols	Their study did not take into account issues such as privacy, smart contract security, trustworthy oracles, scalability, consensus protocols, standardization, interoperability, quantum computing robustness, and governance.
Blockchain technology	A thorough analysis of BC	In this study, BC architecture and fundamental aspects of BC were covered.	In this study, BC architecture and fundamental aspects of BC were covered.
Decentralized FL framework created in BC	A distributed FL architectural context that was created on BC with committee consensus (BFLC)	The quantity of consensus computation may be efficiently reduced with the use of a novel committee consensus technique, which may also lessen malicious attacks.	Time complexity was nottaken into account.
For the allocation of sensitive data in the Internet of Things, BC, and FL	Using blockchain technology, construct a secure data sharing architecture for numerous distributed participants.	The authors added FL to the permissioned BC consensus process, enabling the use of the consensus computing effort for federated training.	The study's technological resources were insufficient.
Vehicle-based edge computing	A secure peer-to-peer data-exchange strategy for in-vehicle computing and systems was proposed.	A secure peer-to-peer data exchange strategy for in-vehicle computing and systems was proposed.	Limited dataset was employed in this investigation.
A federated learning methodology based on blockchain	To maintain the privacy and security of loHT data, FL, and discrepancy confidentiality (DC) were proposed, enabling isolated loHT data to be educated at the holder's location.	The topic of integrating a minimal security and privacy solution into the FL environment was addressed by the authors.	The settings for the accuracy and loss measures are quite low, but they can be raised in the future.

3. Proposed Model

In this section, a novel method for cloud-based smart contracts in FinTech data and their intrusion detection with a cyber-threat federated graphical authentication system are proposed and discussed. Federated learning is used to train models by distributing existing training models to multiple personal edge devices as well as utilizing the edge devices' local data, as shown in Figure 1. In order to update global model, central server accepts as well as incorporates training results from each edge device.

Anomalies in statistics are data points that deviate significantly from other observations (also known as outliers and abnormalities). We consider the regions N1, N2, and N3 to be typical data instance regions since we presume that they are made up of the majority of observations. Data points O1 and O2 can be categorized as anomalies if they are far from these zones. In order to more precisely describe anomalies, we suppose that an n-dimensional dataset with the dimensions I 1, . . . , n, and μ_j follows a normal distribution with mean μ_j and variance σ_j for every dimension. In particular, using the normal distribution assumption, we have the following by using Equation (1):

$$\mu_j = \sum_{i=1}^{m} x_{i,j}/m, \sigma_j^2 = \sum_{i=1}^{m} \left(x_{i,j} - \mu_j\right)^2/m \quad (1)$$

If a new vector x exists, probability $p(x)$ of an anomaly is computed using the formula in Equation (2):

$$p\left(\vec{x}\right) = \prod_{j=1}^{n} p\left(x_j; \mu_j, \sigma_j^2\right) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{\left(x_j - \mu_j\right)^2}{2\sigma_j^2}\right) \quad (2)$$

As shown on the right side of Figure 2, ENs distribute the global method to participants at the beginning of each global iteration, and each fading block of collaborative method download has two phases. NDs and RDs receive the global model from ENs in the first phase. RDs send the received model to FDs in the second phase. Privacy information could be leaked if RDs were able to decode the ENs' superimposed code more quickly than FDs.

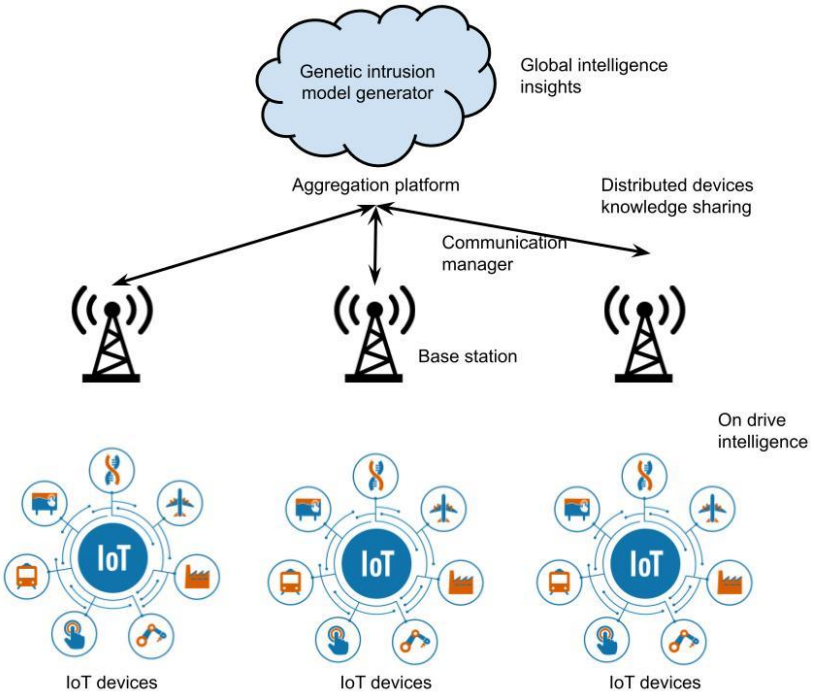

Figure 1. Proposed intrusion detection system for FinTech based on IoT.

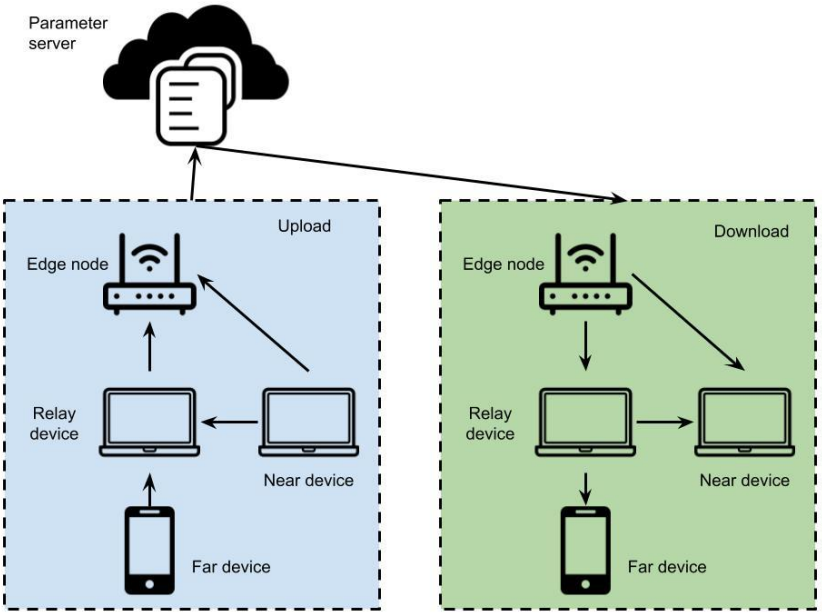

Figure 2. Secure FinTech data analysis.

3.1. Cloud Based Smart Contract Analysis

In order to simplify things, Figure 3 shows how Ethereum smart contracts work without the mining process. The EVM-1 transaction for this smart contract is broadcast to the blockchain after it has been machine-compiled into byte code, where each byte denotes a single operation. A miner intercepts it and confirms Block 1. EVM-2 retrieves web-based data after a user submits a request via a web interface, inserts it into a transaction, and then deploys it to the blockchain. In Block2, the status of transaction tx is modified. Node 3 must synchronize up to at least Block 2 in order to examine the future states contained in the contract and see the changes that tx results in.

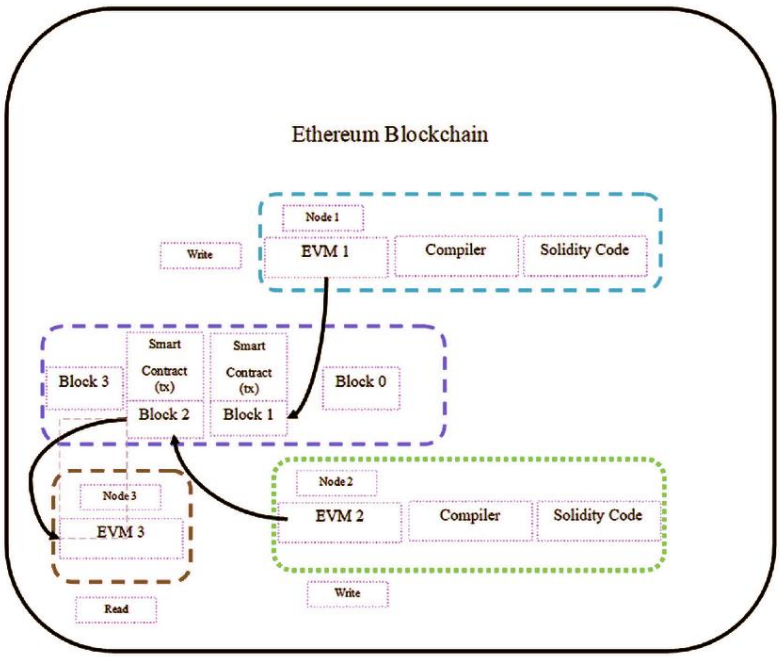

Figure 3. Ethereum Smart Contract Mechanism.

Contracts that we create include metadata about record ownership, permissions, and data integrity. Instructions for managing these properties are cryptographically signed in the blockchain transactions of our system. Before granting third-party viewing permission, a policy may, for instance, require sending of separate consent transactions from healthcare professionals as well as patients.

We planned a framework in view of blockchain shrewd agreements for complex medical services work processes. The management of data access permissions between various entities in healthcare ecosystem is the purpose for which smart contracts are developed for various medical workflows. A number of stakeholders are participating in this plan and engage in a variety of activities, as shown in Figure 3. As can be seen, a smart contract that is kept using blockchain technology might be created to meet all the requirements, from controlling access to data to managing different permissions. This will lead to improved communication between patients and physicians. Data authorization requirements are part of smart contracts. It can also assist in tracking every activity with a unique identifier, from its beginning to its completion. Along with all of the processes and functions that are embedded in the smart contracts, numerous scenarios have been designed as well as explained. Because the operation can be directly managed through smart contract, there will be no need for a centralized entity to manage as well as approve it. The expense of managing the process administratively will be greatly decreased as a result.

We use smart contracts on Ethereum blockchain to record patient–provider connections and add a cryptographic hash of record to blockchain to prevent data manipulation. These contracts link a medical record to data retrieval instructions and viewing rights for remote server execution. Individuals can provide their permission for their medical providers to

share their records, and doctors can add a new record for a particular patient. Both times, the individual receiving the new data is automatically notified and given the option of studying the proposed record before accepting or rejecting it. Participants in the creation of their records thereafter continue to be involved and informed. By offering a specified contract that compiles references to all patient–provider connections for a user, this system prioritizes usability. Users now have a central location to check for updates to their medical history. We use a DNS-like technology to manage identity verification, mapping the user's Ethereum address to an existing and well-known form of identification, such as their name or social security number. After our database authentication server consults the blockchain to confirm permissions, a syncing mechanism manages "off-chain" data transfers between a patient database and a provider database.

3.2. FinTech Intrusion Detection Based IoT Module Using Cyber Threat Federated Graphical Authentication System

Traditional distributed deep-learning algorithms gather and examine a specific volume of private data on central servers during the method training phase, utilizing the distributed stochastic gradient descent (DSGD) method. Because of this, the training process may expose IIoT devices to data privacy leakage risks. There are three phases to the FL procedure: the update phase, aggregation phase, and initialization phase. We suppose FL has N edge devices, and that, during the setup phase, a parameter aggregator, also referred to as a cloud aggregator, distributes a pre-trained global method t based on public datasets to every edge device. To do so, local loss function to be optimized is described as Equation (3):

$$\min_{x\in\mathbb{R}^d} F_k(x) = \frac{1}{D_k}\sum_{i\in D_k} \mathbb{E}_{z_i\sim D_k} f(x; z_i) + \lambda h(x), \tag{3}$$

The cloud aggregator uses Federated during the update phase to obtaina new global model for iteration $t + 1$; therefore we have by Equation (4)

$$\omega_{t+1} \leftarrow \omega_t + \frac{1}{n}\sum_{n=1}^{N} F_{t+1}^n \tag{4}$$

where $\sum_{n=1}^{N} F_{t+1}^n$ denotes method updates aggregation and $\frac{1}{n}\sum_{n=1}^{N} F_{t+1}^n$ denotes average aggregation. The above procedure is repeated by cloud aggregator as well as edge devices until global method converges. We have the following for a deep neural network's fully connected (FC) layer: $b = f(Wa + v)$, where b is output, f is nonlinear mapping, a is input, v is bias, and W is weight. A neural network's most fundamental operation is this formula. The aforementioned formula can be simplified as follows for each specific neuron i:$b_i = \mathrm{ReLU}\left(\sum_{j=0}^{n-1} W_{ij}a_j\right)$, where the activation function is ReLU. The equivalent formula is given as Equation (5) because gradient compression transforms the appropriate weight matrix into a sparse matrix.

$$b_i = \mathrm{ReLU}\left(\sum_{j\in X_i\cap Y} \mathrm{Sparse}\left[I_{ij}\right] a_j\right) \tag{5}$$

where I_j stand for the location data of gradient in weight matrix W and $\sum_{j\in X_i\cap Y} S\left[I_{ij}\right]$ denotes the compressed weight matrix. The communication overhead is decreased by this strategy by saving the gradient in the weight matrix W. Initially, based on load state of all optional edge devices taking part in FL, we determine the load center value of various network resources.

In condition 2, the estimation method is shown, and the heap balance is then identified as Equation (6).

$$l_m = \frac{1}{k}\sum_{i=1}^{k} num\_re_i^m \tag{6}$$

Among these, num rem I signifies number of m resources accessible on edge device I and k denotes number of optional edge devices. Equation (7) illustrates how to evaluate degree of load balance, and Equation (6) can be used to determine it.

$$lb = \frac{1}{k}\sum_{m=1}^{\xi}\sum_{i=1}^{k}\left(l_m^i - l_m\right)^2 \tag{7}$$

After method is supplied to edge device, we measure the edge device's real-time load in accordance with Equation (8).

$$\varepsilon_{ji} = \sum_{m=1}^{\xi} w_{ji}^m\left(1-\mu_{ji}\right) \tag{8}$$

where memory resources (m = 1), CPU resources (m = 2), and bandwidth resources (m = 3) are indicated. The weight of every resource after Model J is sent to the edge device is *wmji*. As seen in Equation (9), *i. mji* represents the quantity of resources that are being used.

$$\mu_{ji}^m = \frac{amo\_re^m}{num\_re_i^m} \tag{9}$$

The number of m resources needed for method training is indicated by notation amo rem. The method for calculating num rem I is given in Equation (10).

$$\text{num\_re} = \text{tot\_re}_i^m - am\_\_re^m, \tag{10}$$

Because edges would be unsightly and would limit the user's ability to select which photographs to click, our system does not adopt Jansen's solution for the graphical passwords. The user was given the option to choose their chosen image. Regardless of size, the technology separates the image into thirty thumbnail photos that are all the same size—10 mm by 10 mm. The frame measures 50 by 60 mm. Although the sizes of mobile touch displays vary, the human–computer interface in our method is uniform in size. The user chooses three to six thumbnails from the image using the touch panel on the mobile device. These pictures are used to create the user's graphical password. The times when the client must input their graphical passwords are indicated in the message section at the base. When a user enters their graphical passwords successfully, this paper vibrates to let them know. Every time a user touches a touchscreen on a portable handheld device, the system records a pressure feature at thumbnail photo.

The system verifies the user's graphical password sequence against the one that was registered during the enrolment process. If the user's login request is inconsistent, the system rejects it. If not, related features are looked at. The user's login is then approved or denied by the system according to a threshold t. The user is valid if $D = t$. If not, the system declines the user's login request. In other words, a person can only log into a system if they can successfully authenticate using both the KDA and graphical password authentications. Thus, even if a shoulder-surfing assault manages to obtain the password, the likelihood of the authentication being compromized is decreased.

$$D = \frac{1}{4k-2}\sum_{f=1}^{4k-2}\frac{X_{v,f}-\mu_f}{\sigma_f} \tag{11}$$

Only the distance between the user's login sample and the training samples is examined in this phase. Regardless of whether the quantity of clients builds, the ideal opportunity for completing the verification will notbe impacted. In a similar vein, computation during this phase is quick. Using the framework of cyber-attack attribution, analysis method, platform construction, and analysis content are all discussed. Additionally, this framework can serve as a model for actual deployment schema design.

The structure of attribution for cyberattacks is depicted in Figure 4. There are three main components to the framework: the beginning of the investigation, attribution analysis,

and threat intelligence, The following sections discuss the functionalities and internal components of each component of the framework: (1) The beginning of the analysis. Based on previous experience with emergency response and cyber-attack analysis, the primary components of the initial data are as follows: samples of malware, information from the network, and logs. We can use a web spider to acquire malware from malware sample websites during the experiment. Malicious activities of malware, such as Cuckoo and ZeroWine, can be examined using the malware sandbox. Network traffic analysis focuses primarily on traffic detection and analysis. To catch bad behavior, we can add maliciousIP addresses and domain names to a blacklist. Traffic analysis can be used to determine the relationship between the traffic data. Wireshark, Moloch, Malcon, and Maltrail, among others, are typical examples of traffic detection and analysis software. Management and evaluation of logs are two of the log-related tasks. Users' access histories, alarm data, operating records, and other data may be included in the log records. Effective log analysis can be provided by robust log management. Malware tests, network traffic, and log records are the beginning of digital-assault attribution examination.

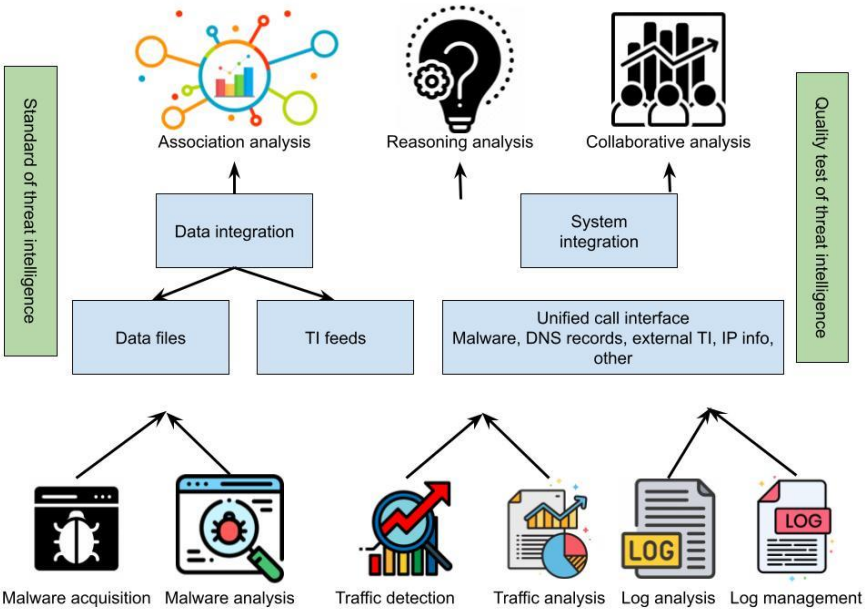

Figure 4. Framework of cyber attack attribution.

(2) Threat intelligence. A threat intelligence quality test, data integration, and system integration are among the duties related to threat intelligence. Common standards for threat intelligence include STIX, TAXII, CybOX, Yara, and OpenIOC, among others. These standards can be utilized and referenced in practical attribution analysis work. The process of combining different data files as well as feeding threat intelligence data to the central database is known as threat intelligence data integration. The IP-related data, including systems such as ZoomEye, Shodan, and IVRE, are combined. In attribution analysis, we can fully utilize threat intelligence through system integration. The purpose of the quality test on threat intelligence is to improve the analysis result by assessing the quality of threat intelligence since its exchange. The attribution of cyber-attacks is based on threat intelligence.

(3) The input for attribution analysis consists of data from threat intelligence. Methods for attribution analysis fall into three categories: reasoning analysis, collaborative analysis, and association analysis. To obtain as much relevant and important data from the threat intelligence database as possible, association analysis is used. During the process of association analysis, primary concerns are efficiency and constraint. Reasoning investigation is used to obtain conceivable relationships and assault chains from related information. In attribution analysis, the goal of collaborative analysis is to make full use of computer performance and analyst thinking. The primary step in attributing a cyberattack is analysis.

The architecture of cyber-attack attribution is introduced into the framework of analysis and threat intelligence. We can learn about the process of attributing cyber attacks as well as the systems and data that are associated with them from the framework. In parallel, we are able to quickly construct a testing environment in accordance with the framework to calculate effort of cyber-attack attribution.

To ensure that a strong D2D connect is laid out, we must also take into account the space constraints while selecting edge devices. Device A will function as a D2D transmitter to connect to the central server if Device A uses D2D communication to communicate with Device B. The SINR of the central server needs to be greater than a predetermined level. As seen in the accompanying example, this value guarantees trustworthiness in multi-hop communication (12).

$$\gamma_{de_a,se} = \frac{\left|h_{de_a,se}\right|^2 P_{de_a}}{\left|h_{me_c,se}\right|^2 P_{me_c} + \sigma^2} \geq \gamma_{de}^{h} \tag{12}$$

Hmec, *se* stands for channel gain between multiplexed cellular user and central server, while *hdea, se* stands for channel gain between Device A and the server. The abbreviations *Pmec* and *Pdea* represent the transmit powers of the respective cellular users and devices, respectively. *De*, which is the SINR of the channel between Device A and the central server, represents the threshold limit that needs to be attained. The SINR threshold is the lowest threshold that satisfies task delay, as can be observed from Equations (13) and (14),

$$R_{de}^{D2D} = B\log_2[1 + \min(\gamma_{de,se})], \tag{13}$$

$$\gamma_{de}^{hh} = 2^{\frac{k_{de}^{D2D}}{s}} - 1 \tag{14}$$

where B is the *D2D* communication link's bandwidth. The interference-to-noise ratio between the apparatus and the central server is represented by γde, se. Equation (15) can be used to represent the minimal transmit power needed for equipment a and the central server to maintain dependable transmission.

$$P_{de_a}^{\min} = \frac{\gamma_{de}^{hh}\left(\left|h_{me_c,se}\right|^2 P_{me_c} + \sigma^2\right)}{\left|h_{de_a,se}\right|^2} \tag{15}$$

The final edge device is reliably chosen when it is based on social attribute perception, which also successfully prevents edge device withdrawal from training and data contamination. All particles are drawn to the average of $P_{best,i}^{t}$and G_{best} when c 1 = c 2.

The examination of the one-dimensional parameter Equations (16) and (17)

$$\mathrm{d}S = \mu(t,\theta,S)\mathrm{d}t + \sigma(t,\theta,S)\mathrm{d}B(t). \tag{16}$$

$$G(\theta) = g_0(S_0 \mid \theta)\prod_{n=0}^{N-1} g(S_{n+1} \mid S_n;\theta) \tag{17}$$

Similarly, Equations (18)–(20) illustrate how to minimize the detrimental effect of the log-likelihood function to estimate θ.

$$-\log G(\theta) = -\log[g_0(S_0 \mid \theta)] - \sum_{n=0}^{N-1}\log[g(S_{n+1} \mid S_n;\theta)] \tag{18}$$

$$S_{n+1} = S_n + \mu(t,\theta,S_n)h + \sigma(t,\theta,S_n)\Delta B_n \tag{19}$$

$$\frac{1}{\sigma(t,\theta,S_n)\sqrt{2\pi h}}\exp\left[-\frac{(S_{n+1} - S_n - \mu(t,\theta,S_n)h)^2}{2\sigma^2(t,\theta,S_n)h}\right] \tag{20}$$

The method of moments achieves the simplest form on a discrete maximum likelihood function by using the approximated PDF indicated above rather than the actual transitional PDF $g(S_{n+1} \mid S_n \mathrm{t}; n\theta)$ in Equations (21)–(23).

$$\overline{\alpha}\left(\overline{\beta}\sum_{n=0}^{N-1} h - \sum_{n=0}^{N-1} S_n h\right) = S_N - S_0 \tag{21}$$

$$\overline{a}\left(\overline{\beta}\sum_{n=0}^{N-1} \frac{h}{S_n} - \sum_{n=0}^{N-1} h\right) = \sum_{n=0}^{N-1} \frac{S_{n+1} - S_n}{S_n} \tag{22}$$

$$\overline{\sigma}^2 = \frac{1}{N}\sum_{n=0}^{N-1} \frac{\left(S_{n+1} - S_n - \overline{\alpha}\left(\overline{\beta} - S_n\right)h\right)^2}{S_n h} \tag{23}$$

Here, we will outline our approach to inferring Markov network structure from data. We develop a probabilistic decision tree for each variable *Xi* to express the conditional probability of that variable given all other variables, *P*(*Xi* | *X Xi*). A set of conjunctive characteristics that can reflect the same probability distribution as the tree is created for each tree. All features are combined into a single model at this point, and any common weight-learning algorithm is used to learn weights worldwide.

With by Equations (24) and (25)

$$\mathcal{F} = \left\{(\boldsymbol{a}, \boldsymbol{\mu}, \boldsymbol{C}) \in \mathbb{R}^{p|\tau_B|} \times \mathbb{R}^{|\tau_B|} \times \mathbb{R}^{K|\tau_L|} : \sum_{k=1}^{K} C_{kt} = 1, C_{kt} \geq 0, \forall k = 1, \ldots, K \forall t \in \tau_L\right\}. \tag{24}$$

$$P_{it}(\boldsymbol{a}, \boldsymbol{\mu}) = \prod_{t \in \mathcal{N}_t^{\text{left}}} p_{it}(\boldsymbol{a}_{\cdot t}, \mu_t) \prod_{t \in \mathcal{N}_t^{\text{right}}} (1 - p_{it}(\boldsymbol{a}_{\cdot t}, \mu_t)), i = 1, \ldots, N, t \in \tau_L \tag{25}$$

For each $k = 1, \ldots, K$, the chance of an individual *II* being placed in class k is equal to $\sum_{t \in \tau_L} P_{it}(\boldsymbol{a}^*, \boldsymbol{\mu}^*) C_{kt}^*$. Probability of belonging to class k returned by randomized tree is equal to Equation (26) for an entering individual with predictor vector x.

$$\mathbf{x} \to \Pi_k(\mathbf{x}) := \sum_{t \in \tau_L} P_{\mathbf{x}t}(\boldsymbol{a}^*, \boldsymbol{\mu}^*) C_{kt}^* \tag{26}$$

The following unrestricted issue would arise according to Equation (27):

$$\begin{gathered} \underset{(a,\mu,\widetilde{a},\widetilde{\mu}) \in \mathbb{R}^{(p+1)(|\tau_B|+|\tau_L|}}{\operatorname{minim}} \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{t \in \tau_L} P_{it}(\boldsymbol{a}, \boldsymbol{\mu})\left(\widetilde{\boldsymbol{a}}_t^{\top} \boldsymbol{x}_i + \widetilde{\mu}_t\right) - y_i\right)^2 \\ + \lambda^{\text{local}} \sum_{j=1}^{p} \left\| \left(\boldsymbol{a}_{j\cdot}, \widetilde{\boldsymbol{a}}_j\right) \right\|_1 + \lambda^{\text{global}} \sum_{j=1}^{p} \left\| \left(\boldsymbol{a}_{j\cdot}, \widetilde{\boldsymbol{a}}_j\right) \right\|_\infty \end{gathered} \tag{27}$$

Using a nonparametric kernel density technique, a closed-form solution generated from the transitional density (28) and (29) can be employed as

$$\overline{g}_M[(t, \mathbf{S}_n) \mid (t_{n-1}, \mathbf{S}_{n-1}); \theta] = \frac{1}{\mathrm{MD}}\sum_{i=1}^{M} K\left(\frac{\mathbf{S}_n - \mathbf{T}_i}{D}\right) \tag{28}$$

$$g[(t_n, \mathbf{S}_n) \mid (t_{n-1}, \mathbf{S}_{n-1}), \ldots, (t_0, \mathbf{S}_0); \theta] \tag{29}$$

Network Training:

To develop a neural network model that depicts the conditional distribution of inputs (for example, an input vector $x = (x_1, x_2, \ldots, x_a)$) to outputs (for example, a class y), one FL challenge is to extract the features from high-dimensional data. Each hidden layer node in this common design computes a weighted average of the inputs after receiving the output of the neurons from the previous layer as input. A non-linear activation function value is ultimately output to the entire input value. A neural network's weight learning issue is a non-linear optimization problem. In supervised learning, the output error of the neural network after passing all training data serves as the objective function. Most of the methods that are utilized to tackle this problem are gradient descent variations. Typically, a random point marks the start of the steep drop. A trainer uses training data to determine

the gradient of the non-linear objective function being optimized in subsequent iterations, updating the weights as necessary. Until the algorithm converges to a local optimum, this procedure will continue for a few epochs. The n participants asynchronously exchange their calculated gradient results to updatethe model parameters with one another after each cycle of training. The shared gradients are entirely within the participant's control. The total of all gradients that went towards the local model's optimum determines the global descent. As a result, participants gain from one another's training datasets and create a more accurate model that can be trained independently and is not constrained to a local training dataset.

Algorithm for Proposed FL-based FinTech:

Initialize $\mathbf{w}, \vartheta_s, \vartheta_n, \vartheta_r, \vartheta_f, \gamma, \eta, \epsilon$

for Incident $e \in 1, \ldots, E$ do
for $b \in B$ do
Rearrange w^0, S_b^0
Reset replay buffer $\mathcal{D}_{b,s}, \mathcal{D}_{b,n}, \mathcal{D}_{b,r}, \mathcal{D}_{b,f}$.
$\vartheta_{b,s}^e = \vartheta_s^e, \vartheta_{b,n}^c = \vartheta_n^e, \vartheta_{b,r}^c = \vartheta_r^e, \vartheta_{b,f}^c = \vartheta_f^e$.
for $k \in 1, \ldots K$ do
In every resource block b, every EN gives S_b^k and updates its observation to $o_{b,s}^k$.
Input $o_{b,s}^k$ into every EN's policy $\vartheta_{b,a}^e$ and finds present pricing method P_b^k

for end devices.

End devices viewer S_b^k and update their observations to $o_{b,n}^k, o_{b,r}^k, o_{b,f}^k$
Input $o_{b,n}^k, o_{b,r}^k, o_{b,f}^k$ to actor network $\vartheta_{b,n}^\epsilon, \vartheta_{b,r}^\epsilon, \vartheta_{k,f}^\epsilon$ and find transmit power $\rho_{b,n}^k, \rho_{b,r}^k, \rho_{b,f}^k$ and jamming coefficient $\Omega_b^{k,d}, \Omega_b^{k,u}$
End devices upload $\mathbf{w}_n^{k+1}$ to specification server while near devices allocate jamming signals.

End devices do τ_b^k round local method update.

PS combined global method for end devices with $\mathbf{w}^{k+1} = \frac{\sum_{n=1}^{N} Q_n \mathbf{w}_n^{k+1}}{Q}$
Updates S_b^k into and evaluates rewards $R_{b,s}^k, R_{b,n}^k, R_{b,r}^k, R_{b,f}^k$
Store transitions $\left(o_{b,s}^k, a_{b,s}^k, R_{b,s}^k, o_{b,s}^{k+1}\right)$ in $\mathcal{D}_{b,s}$

4. Results and Discussion

The model is evaluated with the help of the NSL-KDD dataset. In our research, each client trains a method utilizing local data and uploads updated method parameters to the server for aggregation, allowing the server to dynamically choose the client count. This experiment makes use of a server with the Windows 10 operating system and an Intel(R)Core(TM) i5- 10210U CPU@2.11 GHz processor. Pytorch, a Python deep-learning library, is what we use.

Description of dataset: For the purpose of conducting experiments on intrusion detection in computer networks, Diro and Chilamkurti utilized three original-size datasets known as KDDCUP99, ISCX, and NSL-KDD. They proposed a conveyed deep-learning-based IoT/mist network assault recognition framework, and trials showed fruitful reception of computerized reasoning to online protection purposes. In addition, the distributed architecture attack detection system for IoT applications, such as smart cities, was designed as well as implemented by the authors. To demonstrate the superiority of deep over shallow methods, the evaluation process took into account performance metrics, such as accuracy, detection rate, and false alarm rate. The experimental section looked at two-class and four-class categories in the first round of experiments. In addition, unseen test data were selected as representations of zero-day attack detections.

4.1. Proposed Analysis

The above Table 2 shows the parametric analysis based on the proposed technique for cybersecurity datasets. Here the datasets analyzed are KDDCUP99, ISCX, and NSL-KDD in terms of accuracy, precision, RMSE, recall, F-measure, AUC, trust value, scalability, and integrity.

Table 2. Parametric analysis of proposed technique based on various cybersecurity datasets.

Dataset	Accuracy	Precision	RMSE	Recall	F-Measure	AUC	Trust Value	Scalability	Integrity
KDDCUP99	89	77	55	65	81	77	55	79	59
ISCX	92	85	59	68	83	79	59	81	63
NSL-KDD	95	88	61	71	85	81	63	83	66

Figure 5 shows the parametric analysis for the KDDCUP99 dataset based on the proposed technique for cybersecurity datasets. Here, the proposed technique attained accuracy of 89%, precision of 77%, RMSE of 55%, recall of 65%, F-measure of 81%, AUC of 77%, trust value of 55%, scalability of 79%, and integrity of 59%.

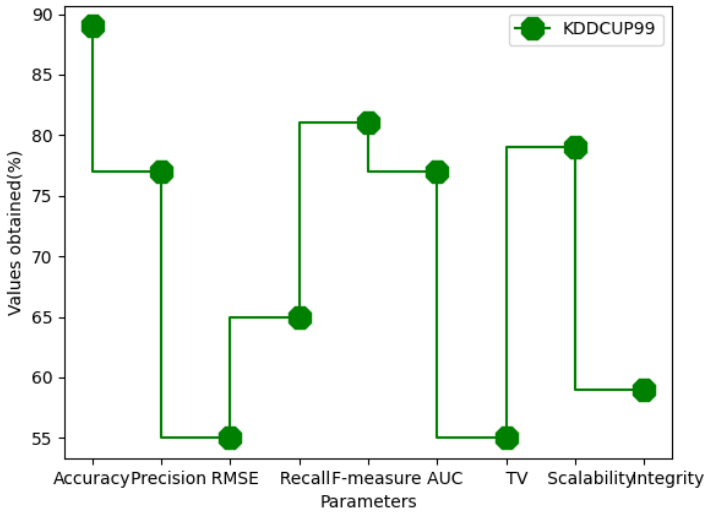

Figure 5. Parametric analysis for KDDCUP99 dataset.

Figure 6 shows the parametric analysis for the ISCX dataset for the proposed technique. The proposed technique attained accuracy of 92%, precision of 85%, RMSE of 59%, recall of 68%, F-measure of 83%, AUC of 79%, trust value of 59%, scalability of 81%, and integrity of 63%.

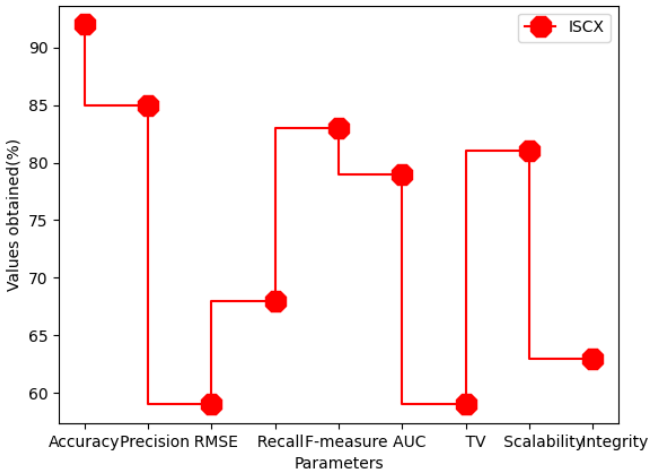

Figure 6. Parametric analysis for ISCX dataset.

Figure 7 shows the parametric analysis for the NSL-KDD dataset based on the proposed technique for the cybersecurity dataset. Here, the proposed technique attained accuracy of 95%, precision of 88%, RMSE of 61%, recall of 71%, F-measure of 85%, AUC of 81%trust value of 63%, scalability of 83%, and integrity of 66%.

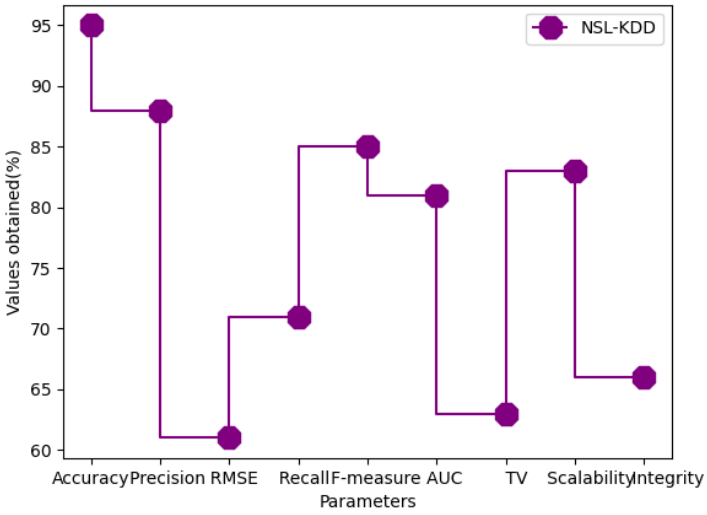

Figure 7. parametric analysis for NSL-KDD dataset.

An effective IDS requires a strong dataset that faithfully captures the reality of real world. In our tests to determine whether the approach outlined in the study is effective, we use a public intrusion detection dataset known as NSL-KDD. The NSL-KDD dataset is extensively utilized for intrusion detection. The NSL-KDD dataset has an advantage over the original KDD CUP 99 dataset in that it doesn't include repeating records in the preparation set, which prevents classifiers from favoring additional continuous records, among other things. This dataset contains the atypical attack types of denial-of-service (DoS), user to root (U2R), remote to local (R2L), and probing attack (Probe). Data about typical network behavior are also supplied. The main emphasis of our work is the classification and detection of four main types of anomalous attacks. Each example in the NSLKDD dataset contains forty-one element credits and one class-recognizable proof, and class ID is utilized to show whether the association record is typical or a particular sort of assault. The system's channel bandwidth is varied between 10 MHz and 50 MHz during the simulations. Equilibrium method is utilized in these simulations to contrast with the proposed method. Under a threshold secure transmission rate, Equilibrium method represents the Stackelberg equilibrium of this multi-agent game. As the channel transfer speed is designated by the framework increment, the general advantage of the proposed approach diminishes. As with distributed ML, we can use this resource to save computation by only performing a small number of local updates instead of a large number of global updates.

4.2. Comparative Analysis

Table 3 shows analyses for various cybersecurity datasets. The datasets analyzed are KDDCUP99, ISCX, and NSL-KDD in terms of accuracy, precision, RMSE, recall, F-measure, AUC, trust value, scalability, and integrity.

Figure 8 shows the analysis of accuracy. Here, the proposed technique attained accuracy of 89%, existing FinTech attained 81%, and MEC attained accuracy of 88% for the KDDCUP99 dataset; for ISCX, the proposed technique attained accuracy of 92%, existing FinTech attained 88%, and MEC attained accuracy of 90%; while the proposed technique attained accuracy of 95%, existing FinTech attained 92%, and MEC attained accuracy of 94% for the NSL-KDD dataset.

Table 3. Analysis based on various cybersecurity datasets.

Dataset	Techniques	Accuracy	Precision	RMSE	F-Measure	AUC	Trust Value	Scalability	Integrity
KDDCUP99	FinTech	81	73	50	75	71	55	77	71
	MEC	88	75	51	78	75	59	79	73
	CSCA_IoT_IFLID	89	77	55	81	77	61	81	75
ISCX	FinTech	88	81	51	80	72	57	82	72
	MEC	90	83	55	81	75	62	83	75
	CSCA_IoT_IFLID	92	85	59	83	79	63	85	79
NSL-KDD	FinTech	92	81	55	79	79	59	85	75
	MEC	94	85	58	81	80	63	89	79
	CSCA_IoT_IFLID	95	88	61	85	81	65	91	83

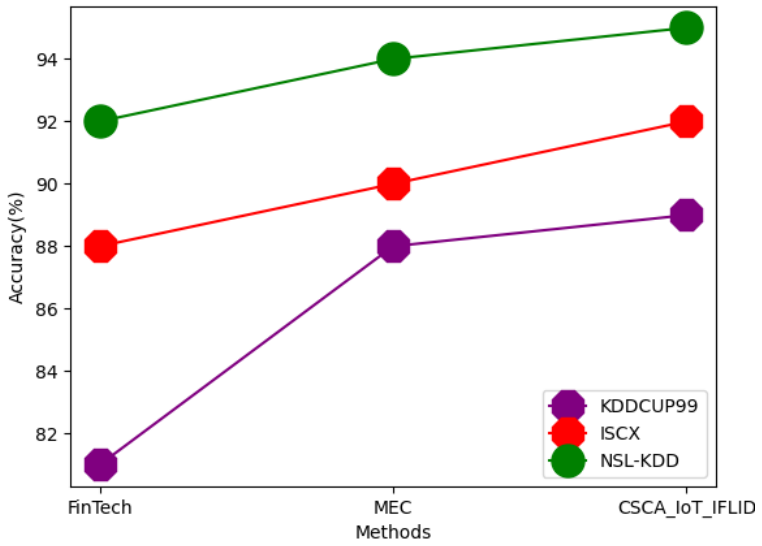

Figure 8. Comparison of accuracy.

Figure 9 shows the comparison of KDDCUP99, ISCX, and NSL-KDD in terms of precision. For the KDDCUP99 dataset, the proposed technique attained precision of 77%, existing FinTech attained 73%, and MEC attained precision of 75%; for ISCX, the proposed technique attained precision of 85%, existing FinTech attained precision of 81%, and MEC attained precision of 83%;while the proposed technique attained precision of 88%, existing FinTech attained precision of 81%, and MEC attained precision of 85% for the NSL-KDD dataset.

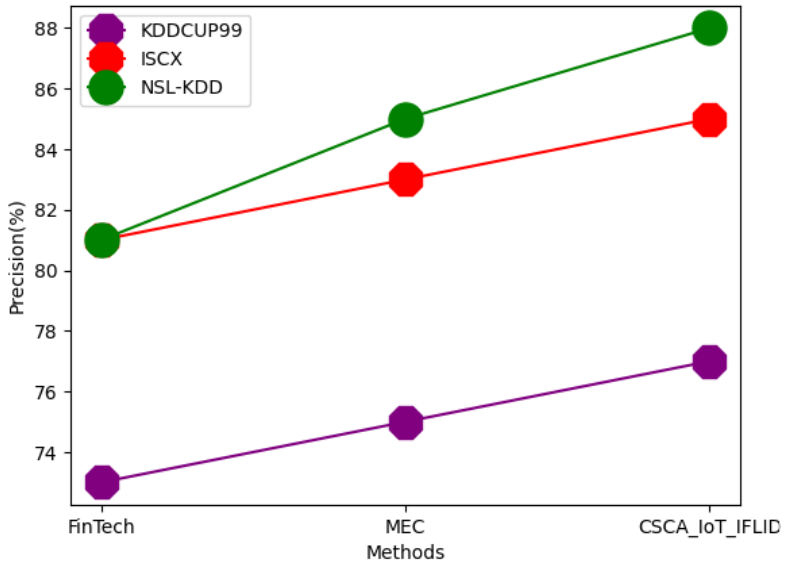

Figure 9. Comparison of precision.

Figure 10 shows the comparative analysis of the RMSE. Here, the proposed technique attained RMSE of 55%, existing FinTech attained 50%, and MEC attained RMSE of 51% for KDDCUP99 dataset; for ISCX, the proposed technique attained RMSE of 59%, existing FinTech attained RMSE of 51%, and MEC attained RMSE of 55%;while the proposed technique attained RMSE of 61%, existing FinTech attained RMSE of 55%, and MEC attained RMSE of 58% for the NSL-KDD dataset.

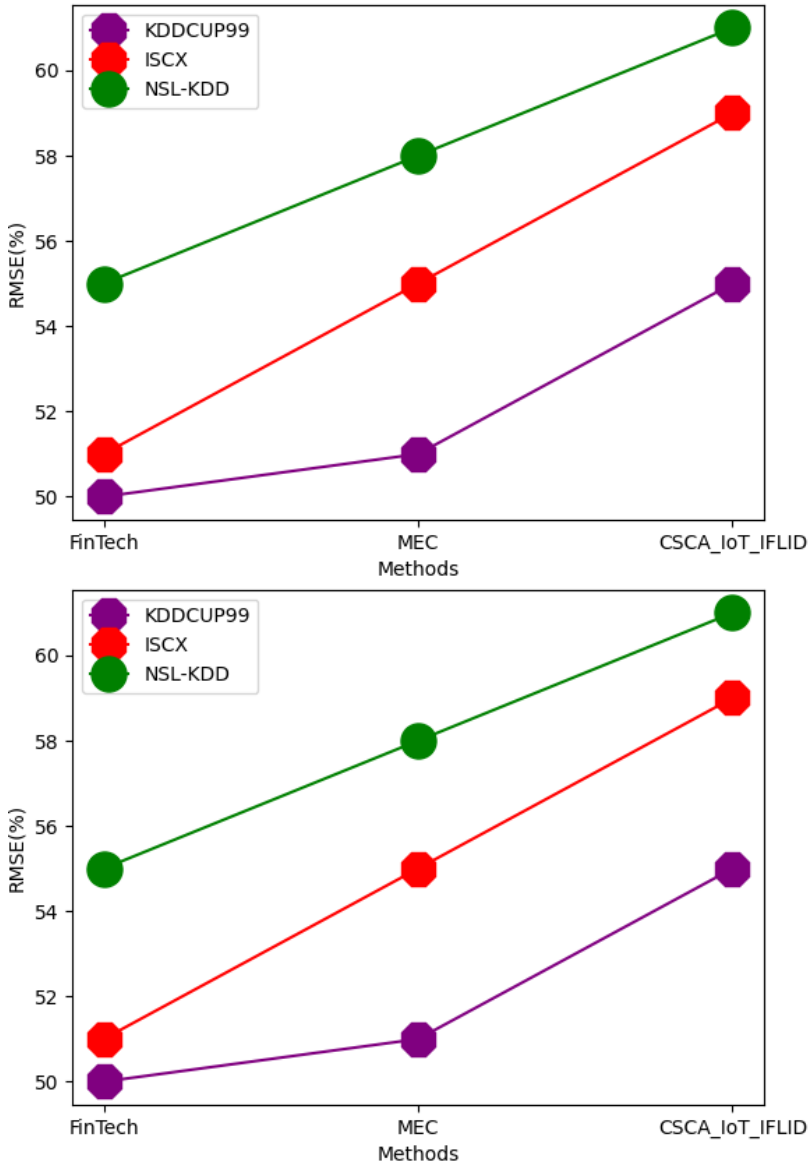

Figure 10. Comparison of RMSE.

Figure 11 shows a comparison of F-measure for KDDCUP99, ISCX, and NSL-KDD. The proposed technique attained F-measure of 81%, existing FinTech attained F-measure of 75%, and MEC attained F-measure of 78% for KDDCUP99 dataset; for ISCX, the proposed technique attained F-measure of 83%, existing FinTech attained an F-measure of 80%, and MEC attained F-measure of 81%; while the proposed technique attained F-measure of 85%, existing FinTech attained F-measure of 79%, and MEC attained F-measure of 81% for NSL-KDD dataset.

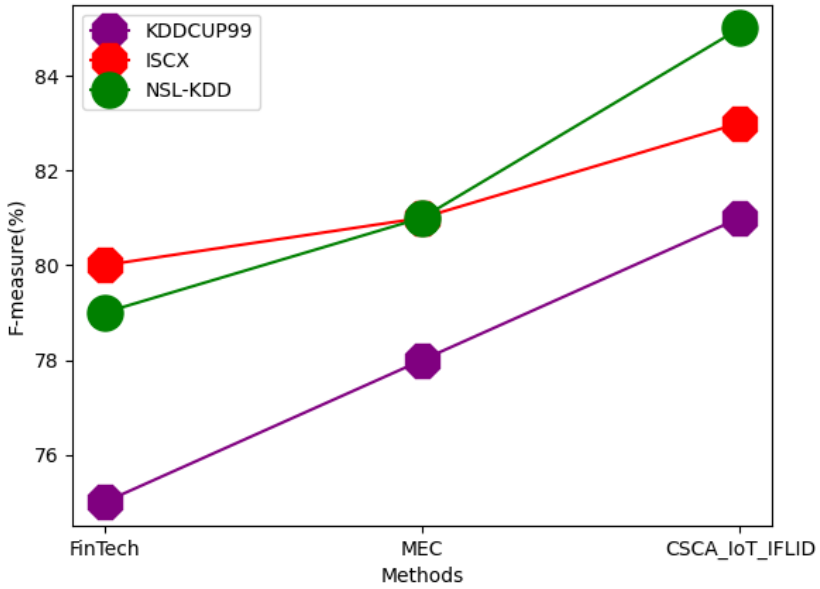

Figure 11. Comparison of F-measure.

Figure 12 shows the comparative analysis of the AUC between the proposed and existing techniques. Here, the proposed technique attained AUC of 77%, existing FinTech attained AUC of 71%, and MEC attained AUC of 75% for KDDCUP99 dataset; for ISCX, the proposed technique attained an AUC of 79%, existing FinTech attained AUC of 72%, and MEC attained AUC of 75%;while the proposed technique attained AUC of 81%, existing FinTech attained AUC of 79%, and MEC attained AUC of 80% for NSL-KDD dataset.

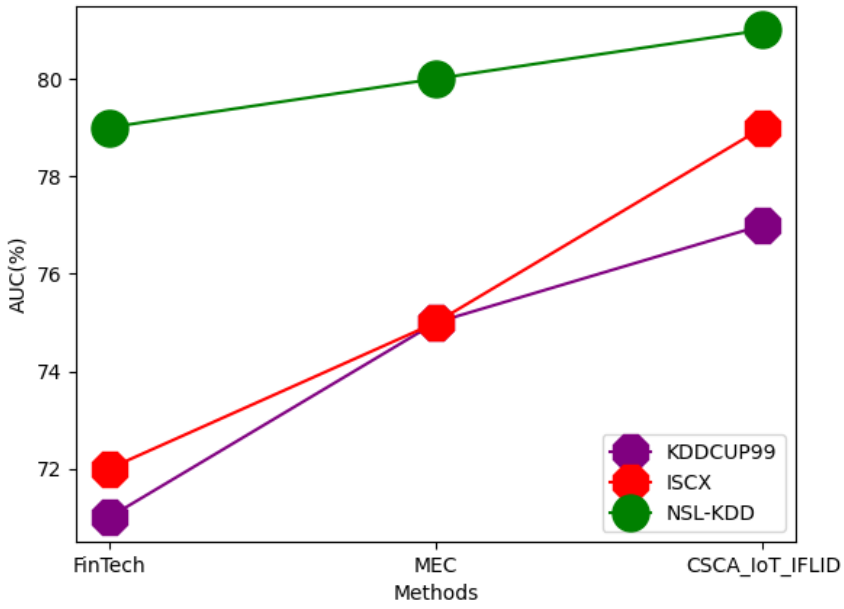

Figure 12. Comparison of AUC.

Figure 13 shows the comparative analysis of the trust value between the proposed and existing techniques. Here, the proposed technique attained a trust value of 61%, existing FinTech attained a trust value of 55%, and MEC attained a trust value of 59% for KDDCUP99 dataset; for ISCX, the proposed technique attained a trust value of 63%, existing FinTech attained a trust value of 57%, and MEC attained a trust value of 62%; while the proposed technique attained a trust value of 65%, existing FinTech attained a trust value of 59%, and MEC attained trust value of 63% for NSL-KDD dataset.

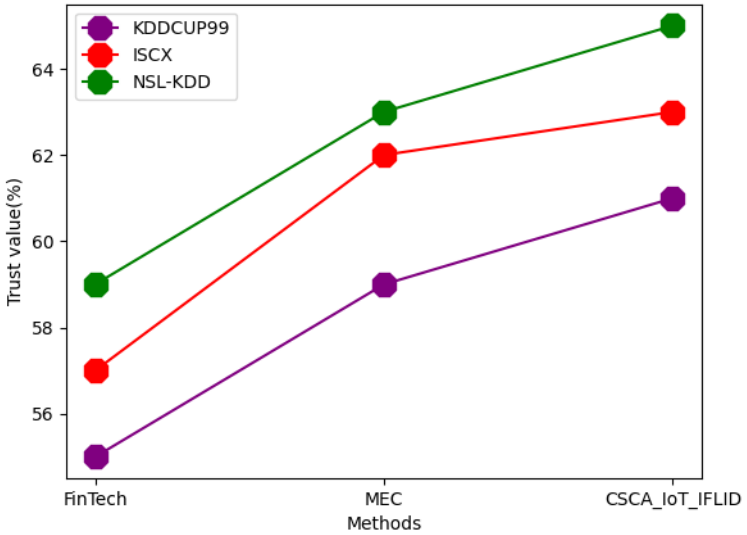

Figure 13. Comparison of trust value.

Figure 14 shows the comparative analysis of scalability between the proposed and existing techniques. Here, the proposed technique attained scalability of 81%, existing FinTech attained trust value scalability of 77%, and MEC attained scalability of 79% for KDDCUP99 dataset; for ISCX, the proposed technique attained scalability of 85%, existing FinTech attained scalability of 82%, and MEC attained scalability of 83%; and the proposed technique attained scalability of 91%, existing FinTech attained scalability of 85%, and MEC attained scalability of 89% for the NSL-KDD dataset.

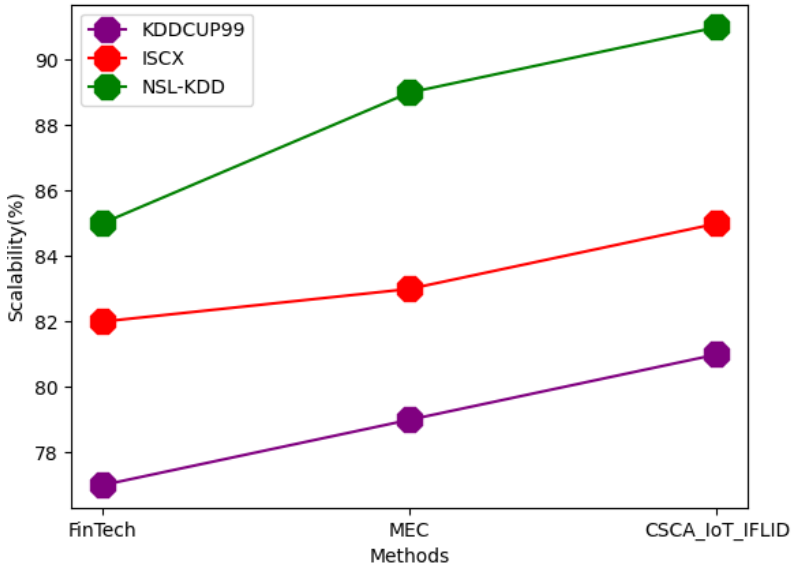

Figure 14. Comparison of scalability.

Figure 15 shows the comparative analysis of integrity between the proposed and existing techniques. Here, the proposed technique attained integrity of 75%, existing FinTech attained integrity of 71%, and MEC attained integrity of 73% for the KDDCUP99 dataset; for ISCX, the proposed technique attained integrity of 79%, existing FinTech attained integrity of 72%, and MEC attained integrity of 75%; while the proposed technique attained integrity of 83%, existing FinTech attained integrity of 75%, and MEC attained integrity of 79% for the NSL-KDD dataset.

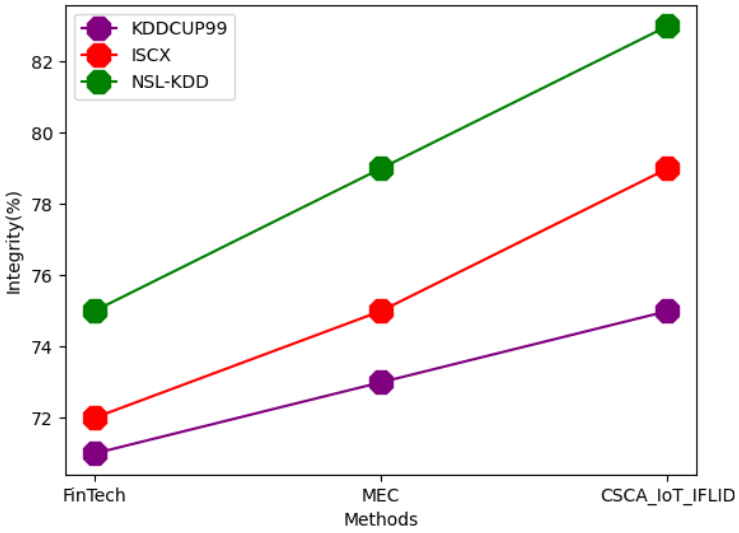

Figure 15. Comparison of Integrity.

4.3. Discussion

Five edge nodes, fifteen end devices, and a parameter server make up the system in the experiment. Each of the five resource blocks contains an edge node, a device that is close by, a device that is relayed, and a device that is far away.

Because the system is set up for full participation, all devices, regardless of whether they have more or fewer data, must participate in model training. For a single resource block, the SNR is 30, the path loss factor is 2, and the channel bandwidth is set to $W = 20$. Default values for data quantity and data quality in simulation experiments are $Q = 512$ and 0.5, respectively. On Windows 10, we use PyTorch 1.8 to carry out our experiments. We utilized a fully connected multi-layer processor with two hidden 64-unit output layers and headers to represent the policy. The actor network server's pricing strategy is determined by three headers. There are three headers for near and far devices and two headers for relay devices in the devices actor network to determine their communication strategies. The value coefficient is $c = 0.5$, the clip ratio is equal to 0.2, the discount factor is equal to 0.99, and local learning rate is set to 0.0003. The number of training episodes is set to $E = 100$, each episode has a length of $K = 16$, and the number of previous experiences is $L = 4$. Depending on its computation resource, each device can select a different Db 16, 32 of the mini-batch size when performing the local update. The neural networks are then optimized using the Adam optimizer. The proposed technique attained accuracy of 89%, precision of 77%, RMSE of 55%, F-measure of 81%, AUC of 77%, trust value of 61%, scalability of 81%, and integrity of 75%.; existing FinTech attained accuracy of 81%, precision of 77%, RMSE of 50%,F-measure of 75%, AUC of 71%, trust value of 63%, scalability of 85%, and integrity of 79%; MEC attained accuracy of 88%, precision of 75%, RMSE of 51%,F-measure of 78%, AUC of 75%, trust value of 65%, scalability of 91%, and integrity of 83%.

5. Conclusions

This study aimed to develop a cloud-based IDS based on IoT federated learning architecture and smart contract analysis. A federated graphical authentication system was developed for cloud-based smart contracts in FinTech data and their intrusion detection with the help of cyber threats. Data augmentation is intended to increase the quantity of training data by making use of knowledge that is already present in local training data. This will increase the local model's generalizability, while failing to generalize to data that has not yet been discovered. For secure certification of credit sharing method results under FL, we suggest authority control contracts and credit verification contracts. According to thorough experimental results and security research, our suggested credit model sharing system, which is based on FL as well as blockchain, is extremely accurate, effective, and

stable. The proposed technique attained accuracy of 95%, precision of 85%, RMSE of 59%, recall of 68%, F-measure of 83%, AUC of 79%, trust value of 65%, scalability of 91%, and integrity of 83%. The proposed method may be used broadly in industries where data security and privacy are crucial, yet cooperation across organizations can result in significantly higher performance and accuracy. We plan to develop class-adaptive solutions in the context of ongoing research.

It is simple to plan correspondence-productive methodologies that convey short messages or model changes over and over as a component of the preparation interaction, as opposed to sending the total informational collection over the organization. Connection, bandwidth, and power are essential for maintaining these activities because this process is carried out in millions of tiny devices. The best two choices for making these cycles more effective and diminishing correspondence in stages are: (1) reducing the total number of rounds of communication, and (2) reducing the number of messages exchanged during each round.

Author Contributions: Conceptualization, V.N.K.; original draft and review& editing, V.J.; validation, M.K.; proposal of the new methodology, M.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset presented in this study is openly available at https://www.unb.ca/cic/datasets/nsl.html, https://www.kaggle.com/datasets/galaxyh/kdd-cup-1999-data; https://www.unb.ca/cic/datasets/ids.html., accessed on 9 January 2023.

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

References

1. Chuang, L.M.; Liu, C.C.; Kao, H.K. The adoption of fintech service: TAM perspective. *Int. J. Manag. Adm. Sci.* **2016**, *3*, 1–15.
2. Hu, Z.; Ding, S.; Li, S.; Chen, L.; Yang, S. Adoption intention of fintech services for bank users: An empirical examination with an extended technology acceptance model. *Symmetry* **2019**, *11*, 340. [CrossRef]
3. Dospinescu, O.; Dospinescu, N.; Agheorghiesei, D.T. *Fintech Services and Factors Determining the Expected Benefits of Users: Evidence in Romania for Millennials and Generation Z*; Technical University of Liberec: Liberec, Czechia, 2021.
4. Nasir, A.; Shaukat, K.; Iqbal Khan, K.; Hameed, I.A.; Alam, T.M.; Luo, S. Trends and directions of financial technology (Fintech) in society and environment: A bibliometric study. *Appl. Sci.* **2021**, *11*, 10353. [CrossRef]
5. Mai, T.; Yao, H.; Xu, J.; Zhang, N.; Liu, Q.; Guo, S. Automatic double-auction mechanism for federated learning service market in internet of things. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3123–3135. [CrossRef]
6. Aouedi, O.; Piamrat, K.; Muller, G.; Singh, K. Federated semisupervised learning for attack detection in industrial Internet of Things. *IEEE Trans. Ind. Inform.* **2022**, *19*, 286–295. [CrossRef]
7. Wan, Y.; Qu, Y.; Gao, L.; Xiang, Y. Privacy-preserving blockchain-enabled federated learning for B5G-Driven edge computing. *Comput. Netw.* **2022**, *204*, 108671. [CrossRef]
8. Noman, A.A.; Rahaman, M.; Pranto, T.H.; Rahman, R.M. Blockchain for medical collaboration: A federated learning-based approach for multi-class respiratory disease classification. *Healthc. Anal.* **2023**, *3*, 100135. [CrossRef]
9. Islam, A.; Al Amin, A.; Shin, S.Y. FBI: A Federated Learning-Based Blockchain-Embedded Data Accumulation Scheme Using Drones for Internet of Things. *IEEE Wirel. Commun. Lett.* **2022**, *11*, 972–976. [CrossRef]
10. Loh, Y.; Chen, Z.; Zhao, Y.; Yu, H. FLAS: A Platform for Studying Attacks on Federated Learning. In *Social Computing and Social Media: Design, User Experience and Impact, Proceedings of the 14th International Conference, SCSM 2022, Held as Part of the 24th HCI International Conference, HCII 2022, Virtual Event, 26 June–1 July 2022, Part I*; Springer International Publishing: Cham, Switzerland, 2022; pp. 160–169.
11. Sheng, X.; Gao, Z.; Cui, X.; Yu, C. Federated Reinforcement Learning Technology and Application in Edge Intelligence Scene. In *Advances in Internet, Data & Web Technologies, Proceedings of the 11th International Conference on Emerging Internet, Data & Web Technologies (EIDWT-2023), Semarang, Indonesia, 23–25 February 2023*; Springer International Publishing: Cham, Switzerland, 2023; pp. 284–291.
12. Lyu, L.; Yu, H.; Ma, X.; Chen, C.; Sun, L.; Zhao, J.; Yang, Q.; Philip, S.Y. Privacy and robustness in federated learning: Attacks and defenses. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, 1–21. [CrossRef] [PubMed]

13. Khadidos, A.; Subbalakshmi AV, V.S.; Khadidos, A.; Alsobhi, A.; Yaseen, S.M.; Mirza, O.M. Wireless communication based cloud network architecture using AI assisted with IoT for FinTech application. *Optik* **2022**, *269*, 169872. [CrossRef]
14. Ali, A.; Almaiah, M.A.; Hajjej, F.; Pasha, M.F.; Fang, O.H.; Khan, R.; Teo, J.; Zakarya, M. An industrial IoT-based blockchain-enabled secure searchable encryption approach for healthcare systems using neural network. *Sensors* **2022**, *22*, 572. [CrossRef] [PubMed]
15. Upreti, K.; Syed, M.H.; Khan, M.A.; Fatima, H.; Alam, M.S.; Sharma, A.K. Enhanced algorithmic modelling and architecture in deep reinforcement learning based on wireless communication Fintech technology. *Optik* **2023**, *272*, 170309. [CrossRef]
16. Kumar, P.; Kumar, R.; Gupta, G.P.; Tripathi, R. BDEdge: Blockchain and deep-learning for secure edge-envisioned green CAVs. *IEEE Trans. Green Commun. Netw.* **2022**, *6*, 1330–1339. [CrossRef]
17. Singh, R.; Deorari, R. Enhancing Collaborative Intrusion detection networks against insider attack using supervised learning technique. In Proceedings of the 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 16–17 October 2022; pp. 1–6.
18. Sheth, H.S.K.; Ilavarasi, A.K.; Tyagi, A.K. Deep Learning, blockchain based multi-layered Authentication and Security Architectures. In Proceedings of the 2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC), Salem, India, 9–11 May 2022; pp. 476–485.
19. Baliker, C.; Baza, M.; Alourani, A.; Alshehri, A.; Alshahrani, H.; Choo, K.K.R. On the Applications of Blockchain in FinTech: Advancements and Opportunities. *IEEE Trans. Eng. Manag.* **2023**. [CrossRef]
20. Ch, R.; Srivastava, G.; Nagasree YL, V.; Ponugumati, A.; Ramachandran, S. Robust Cyber-Physical System Enabled Smart Healthcare Unit Using Blockchain Technology. *Electronics* **2022**, *11*, 3070. [CrossRef]
21. Stojanović, B.; Božić, J. Robust Financial Fraud Alerting System Based in the Cloud Environment. *Sensors* **2022**, *22*, 9461. [CrossRef] [PubMed]
22. Gehlot, A.; Joshi, A. Multilayer Statistical Intrusion Detection Model for Wireless Network. In Proceedings of the 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 16–17 October 2022; pp. 1–7.
23. Rana, A.; Srivastava, V.K. Design of IoT network using Deep learning model for Anomaly Detection. In Proceedings of the 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 16–17 October 2022; pp. 1–8.