A Secure AI Framework for Intelligent Traffic Prediction and Routing in SDN Based Consumer Internet of Things

Zabeehullah, Qazi Mazhar ul Haq*, *Member, IEEE*, Fahim Arif *Senior Member, IEEE*, Muhammad Shahid Anwar, *Member, IEEE*, and Wadee Alhalabi, Nauman Ali Khan,

***Abstract*—With the rapid increase of Consumer Internet of Things (CIoT) and advancements in communication technologies, both are generating a huge amount of imbalance data. Traditional network architectures struggle to handle the complex and heterogeneous nature of CIoT devices, as well as the imbalance and unpredictability of traffic flows. Software Defined Networking (SDN) is a novel networking paradigm. By decoupling the data plane from the control plane, it efficiently manages the complexity and heterogeneity of CIoT devices. However, challenges such as imbalance data security, future traffic load prediction, and optimized routing still persist in CIoT environment. Advancements in Deep Learning (DL) algorithms, along with their extensive application in CIoT domain, have enabled resolving SDN-CIoT security and performance issues. To address the above mentioned challenges, in this article, we propose an AI-based framework which comprises two modules: 1) DL-based security and traffic load prediction module and 2) DRL-based routing optimization module. In addition, the proposed framework employs the CNN based intelligent load balancing strategy among SDN controllers to reduce the computational burden on the main controller. The performance of the proposed model is evaluated through simulation and results demonstrate that the proposed framework achieved excellent performance compared to the state-of-the art methods.**

***Index Terms*—Consumer Internet of Things, Software Defined Networking, Deep Learning, Deep Reinforcement Learning, Routing Protocols, Imbalance Data Security.**

I. Introduction

THE Internet of Things (IoT) integration with conventional Consumer Electronics (CE) has transformed them into next-generation Consumer Internet of Things (CIoT) with increased intelligence and connection [1]. The connectivity of sensors, actuators, appliances, and other consumer products enables better data availability and automatic control in the CIoT network [2].

Smart CIoT devices generate a huge amount of imbalance data with unpredictable traffic flows [3]. The challenge of detecting security attacks and predicting future traffic load worsens when dealing with imbalance and unpredictable incoming flows from CIoT devices [4], [5]. Detecting anomalies and attacks from imbalance data sets is particularly challenging due to the inherent incompleteness of the data, which leads to the omission of important information necessary for accurate attack prediction. Moreover, the CIoT underlying infrastructure is complex, heterogeneous, and insecure, leading to network security and performance issues. This poses a significant challenge for traditional networks in effectively managing the complexity and heterogeneity of CIoT, as well as in providing adequate security measures and performance improvements. The rise of Software Defined Networking (SDN), has contributed to mitigating CIoT heterogeneity, security, and performance challenges to some extent. SDN, owing to its network-wide perspective enables the capability to control plane to be fully aware of the underlying network environment, enabling it to issue periodic instructions to the data plane for enhancing network security and performance [6]. Moreover, the SDN manages the CIoT devices heterogeneity through programmable and centralized controller which abstract the network infrastructure. This abstraction simplifies the management of heterogeneous CIoT devices, making it easier to integrate, scale, and secure them within a unified network. The combination of SDN and CIoT is commonly denoted as SDN-CIoT. The SDN-CIoT framework consists of three layers, namely the sensing-actuator layer, data layer, and control layer, as illustrated in Fig 1. Better opportunities for research into how to enhance security and performance challenges are provided by the SDN-based CIoT system [7].

Despite significant advancements in SDN and DL-based methodologies, several challenges persist in the domain of secure routing optimization and traffic load prediction in SDN-CIoT environments. The existing approaches face the following limitations: Javeed et al. [8] proposed an anomaly detection approach integrating SDN and DL for CIoT. However, their method does not address the challenges of imbalance data security attacks, leading to reduced detection accuracy for minor class attacks. Additionally, their model lacks a robust mechanism for real-time traffic prediction, which is crucial

Qazi Mazhar ul Haq is with the Department of Computer Science and Engineering and International Bachelor Program in Informatics(IBPI) Yuan Ze University Taiwan. email: (qazi@saturn.yzu.edu.tw)

Zabeehullah, Fahim Arif, and Nauman Ali Khan are with the Department of Computer Software Engineering National University of Sciences and Technology (NUST), Islamabad, Pakistan. (email: zabeeh.phdcse@students.mcs.edu.pk, fahim@mcs.edu.pk, nauman@mcs.edu.pk)

Muhammad Shahid Anwar is with the Department of AI and Software, Gachon University Seongnam-si 13120, South Korea, shahidanwar786@gachon.ac.kr

Wadee Alhalabi is with the Department of computer science King Abdulaziz University, Jeddah, Saudi Arabia, wsalhalabi@kau.edu.sa

Corresponding author: Qazi Mazhar ul Haq email:qazi@saturn.yzu.edu.tw, Co-corresponding author: Muhammad Shahid Anwar shahidanwar786@gachon.ac.kr

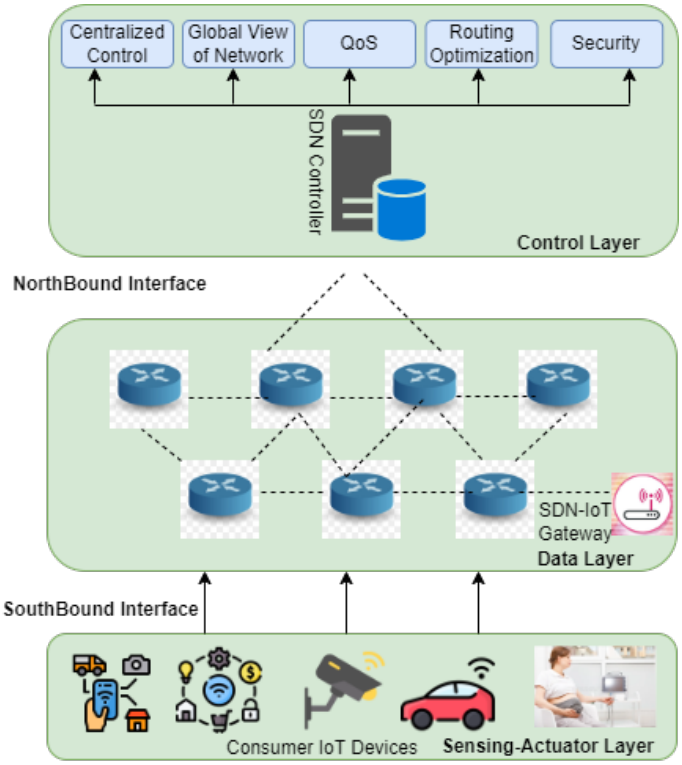


Fig. 1. The SDN-CIoT architecture with three layers: sensing-actuator layer, data layer, and control layer.

for efficient routing in dynamic CIoT environments. The adversarial ML-based secured cloud architecture proposed in [9] improves security in Smart Healthcare CIoT. However, it is primarily designed for cloud-based CIoT environments and does not effectively address the challenges of decentralized SDN-based CIoT networks, such as scalability and real-time attack mitigation. Kim et al. [10] introduced an SDN and SL-based framework for traffic anomaly detection in smart city CIoT devices. While their method enhances traffic monitoring, it lacks a proactive mechanism to optimize routing based on predicted network congestion and security threats. Furthermore, it does not incorporate an adaptive learning model capable of handling dynamic and evolving attack patterns. The scheduling technique suggested in [11] allocates jobs to the nearest MEC server with adequate computational resources in an SDN environment. However, it does not integrate security mechanisms to detect potential threats during job allocation. Additionally, it does not account for network congestion and load-balancing strategies, which may lead to suboptimal resource utilization. The strategy combining blockchain and SDN multi-controller systems proposed in [12] enhances security against various attacks. Nevertheless, it does not explicitly tackle the challenge of intelligent routing optimization in dynamic CIoT networks. Additionally, the high computational cost of blockchain integration can result in increased latency, making it less feasible for real-time applications. Montazerolghaem et al. [13] addressed QoS and QoE challenges in Software-Defined Internet of Vehicles by proposing a modular architecture named ELQ^2. However, their approach primarily focuses on multimedia streaming and does not address security vulnerabilities in SDN-CIoT environments. Moreover, their model does not incorporate intelligent intrusion detection mechanisms to mitigate evolving cyber threats. Guo et al. [14] proposed a DRL-based approach for secure routing optimization in SDN-CIoT environments. However, their method is limited to known attack patterns, making it less effective against zero-day attacks. Furthermore, their approach lacks an adaptive mechanism to handle imbalanced data, which can lead to biased attack detection. An LSTM-based approach for network attack detection in SDN-supported IoT networks was proposed in [15]. While LSTM models perform well in sequential data analysis, their high computational cost and long training times make them unsuitable for real-time attack detection. Additionally, they do not incorporate traffic prediction mechanisms for optimizing routing decisions. Manocha et al. [16] introduced an Optimized Deep Neural Network (ODNN) algorithm for detecting malicious CIoT devices based on energy characteristics. However, their method is highly dependent on predefined energy profiles, making it less adaptable to dynamic attack patterns. Additionally, it does not provide an efficient strategy for routing optimization in large-scale SDN-CIoT networks. Finally, the comprehensive review of DL-based routing optimization algorithms in SDN presented in [17] highlights various methodologies but does not propose a unified solution that simultaneously addresses security threats, traffic load prediction, and routing optimization. Existing approaches often focus on one aspect while overlooking the interdependencies between security, traffic prediction, and routing performance. These limitations emphasize the need for an integrated AI-driven framework that simultaneously enhances security, predicts future traffic loads, and optimizes routing decisions in SDN-CIoT environments.

Most existing literature on secure routing optimization in the SDN-CIoT domain using DL methodologies overlooks the imbalance and dynamic behavior of CIoT traffic. This imbalance causes DL models to favor majority classes, leading to inaccurate identification of minor class attacks. Optimizing routing with dynamic CIoT traffic flows is also challenging task. Despite significant research efforts, gaps remain in addressing anomaly detection, future traffic load prediction, and routing optimization for imbalance and dynamic CIoT data. To tackle these challenges, this article proposed a framework with two modules: one using DL algorithms for handling imbalance data and predicting future traffic loads, and another using DRL for routing optimization. To the best of our knowledge, this is the first effort to address imbalance data security, traffic load prediction, and routing optimization in the SDN-CIoT domain.

The main contributions of this article are as follows:

- A novel hybrid model is introduced that combines Generative Adversarial Networks (GANs) with Convolutional Neural Networks (CNNs) for intelligent attack detection in imbalanced data. This approach enhances security while integrating predictive capabilities for future traffic loads.
- The framework includes a cutting-edge Deep Reinforcement Learning (DRL) module to optimize routing decisions. The DRL model learns dynamically from the

network environment, enabling adaptive and efficient routing that minimizes congestion and optimizes resource utilization.
- A novel CNN-based strategy for load balancing across multiple controllers within the SDN-CIoT network is proposed. This approach intelligently measures the load on each controller and dynamically redistributes it, ensuring balanced performance, preventing single points of failure.

The rest of the paper is organized as follows:

Section II outlines the related work of the proposed framework. Section III presents the problem statement, while Section IV elucidates the workings and architecture of our proposed framework. In Section V, assess the performance of the proposed framework, and Section VI describes the results and discussions. Finally, Section VII concludes the article.

II. Related Work

This section provides a literature review focused on DL-based approaches in SDN-CIoT, specifically examining imbalance data security, traffic load prediction, and routing optimization. Javeed et al. [8] proposed an efficient and intelligent IDS for a network of consumer electronic devices, utilizing both SDN and DL techniques. Popoola et al. [18] exploited the Federated Learning (FL) technique for intrusion detection of CIoT devices. DL driven SDN enabled IDS is proposed to combat emerging cyber threats in IoT [19]. IHSF proposed approach has three solutions which optimally computes forwarding paths. The simulation results show that the proposed IHSF solution has a better performance than the existing approach in terms of end-to-end delay, and packet delivery ratio [20]. The authors of [21] proposed a modular energy and load control solution for the Internet of Medical Things that makes use of virtual resources and network softwarization. The suggested controller first accurately determines the size of the IoMT network in order to dynamically modify the resources. Next, it routes traffic between switches to the desired server and divides the load among the IoMT servers. A novel IHSF model has been proposed which computes optimal path forwarding in hybrid SDN F for time-critical traffic flows generated by IoT applications. Ahmed et al. [22] developed a mechanism for computing the optimal routing or path forwarding without considering the security attacks. Athena as a new SDN-based software solution has been proposed that exports a well-structured development interface and provides general purpose functions for rapidly synthesizing a wide range of anomaly detection services [23]. The authors of [24] proposed SDN and DL based model for imbalance data anomalies detection in IoMT. An IDS is proposed based on the DL architecture to protect the CAN bus in vehicles [25]. Yamauchi et al. [26] proposed a DL-based IDS focusing on user behavior to safeguard smart electronic devices in smart homes. Chuang et al. [27] introduced a routing and forwarding algorithm that takes flow characteristics into account in order to address the issue of SDN scalability in wireless data centers. The authors of [28] proposed an energy management system for smart homes using IoT and big data. DL-based security system called BDHDLS is proposed for detecting attacks and anomalies in traffic flows [29]. The authors of [30] presented a survey focused on the following: First, dissecting the SDN architecture and looking into the load balancing issue within SDN. Second, classifying AI-based load balancing techniques and carefully evaluating these mechanisms from a variety of angles, including the problem addressed, the algorithm/technique used, and their advantages and disadvantages. Thirdly, providing an overview of the metrics used to assess these strategies' efficacy. Lastly, highlighting the developments and difficulties in AI-based load balancing for next studies. Dynamic DRL based task offloading algorithm is proposed to minimize latency and energy consumption in CIoT [31]. Gao et al. [32] introduced an intrusion detection model based on LSTM and FNN, which achieved excellent accuracy and efficiency. Regarding the controller placement issue in SDN, the authors of [33] provided a survey. In order to minimize the packet propagation latency between controllers and switches, they first study the state-of-the-art methods and create a taxonomy depending on their goals. They also list the outstanding issues and ongoing research challenges related to this area in order to stimulate future study. The authors of [34] proposed a DL-based intelligent traffic load prediction and channel assignment techniques in an SDN-IoT environment. Due to the rapid increase in sensory data and the demand for swift responses in the IoT delivery network, ensuring high-speed data transmission has emerged as a pivotal issue. The authors of [35] presented an innovative channel assignment algorithm in SDN-IoT, employing DL based prediction and a partially overlapping channel allocation technique. DRL and SDN based framework has been proposed to ensure security and performance in SDN-IoT domain [36]. Aslam et al. [37] proposed a framework which employed ML techniques within an adaptable multilayered feed-forwarding approach to effectively identify DDoS attacks. The authors of [38] presented a QoS aware SDN-IoT architecture to simultaneously balance traffic between IoT servers and meet the QoS needs of different IoT services. The summary of the related work is shown in a Table I.

III. Problem definition, System Model and its description

A. Problem Definition

In an SDN-CIoT environment, the network traffic primarily consists of genuine flows, with a small proportion being malicious or attacked flows that can disrupt network performance. While most attacks are known and preventable, rare and infrequent attacks pose challenges in detection due to imbalanced data. Basic deep learning models struggle to effectively detect these uncommon attacks because they have difficulty understanding abnormal behavior, resulting in less accurate network attack detection. Addressing this challenge is the focus of the first module in the proposed framework. Another challenge involves predicting future network traffic loads and congestion. Unlike traditional networks where traffic load is stable and continuous, SDN-CIoT environments, particularly in 5G/6G settings, exhibit complex and rapidly changing traffic patterns resembling bursts. The sensing plane

TABLE I
COMPARISON OF THE PROPOSED TECHNIQUE WITH THE RELEVANT TECHNIQUES EXIST IN THE LITERATURE.

Ref.	DL	MSC	Imbalance IoT Data Security	Balance IoT Data Security	TLP	RO
[12]	✗	✓	✗	✓	✗	✗
[14]	✓	✗	✗	✓	✓	✓
[8]	✓	✗	✗	✓	✗	✗
[19]	✓	✗	✗	✓	✗	✗
[20]	✓	✗	✗	✓	✗	✓
[24]	✓	✗	✗	✓	✗	✗
[25]	✓	✗	✗	✓	✗	✗
[26]	✓	✗	✗	✓	✗	✗
[29]	✓	✗	✗	✓	✗	✗
[34]	✓	✗	✗	✗	✓	✓
[35]	✓	✗	✗	✗	✓	✓
[37]	✓	✗	✗	✓	✗	✗
This work	✓	✓	✓	✓	✓	✓

DL = Deep Learning, MSC = Multi-SDN Controller, TLP = Traffic Load Prediction, RO= Routing Optimization

in SDN-CIoT involves heterogeneous CIoT devices categorized by their sensing mechanisms: periodic, event-based, and query-driven sensing. In the SDN base CIoT systems, the diverse nature of CIoT nodes and their lack of cooperation pose a significant challenge for SDN-CIoT enabled switches when attempting to predict the actual future network flows. Based on attack-free and future-predicted traffic flows, the second module of the proposed framework performs DRL-based routing optimization.

B. System Model

We assume that the SDN-CIoT environment consists of heterogeneous CIoT devices and sensors. These devices collect information from their surroundings and transmit these information to the gateway through many SDN-IoT enabled switches. To simplify and improve the understanding of the proposed framework, we have introduced some notations. We define the graph $G = \{D \cup S \cup C, E\}$, D represents the number of devices in the system. S represents the number of switches in the SDN-CIoT system: $S = \{s_1, s_2, ..., s_T\}$, where T represents the total number of switches. It is assumed that switches are randomly deployed and devices are served by the each switch within its designated area . For instance, an Access Point (AP) installed in a house, serving all the CIoT devices and sensors in that house, is considered a switch. The average number of CIoT devices connected to a switch is denoted as R. Therefore, the total number of devices $|D|$ in the system can be defined as $|D| = T \times R$. The primary function of each switch in the system is to gather data from CIoT devices within its coverage range and forward it to the gateway.

IV. PROPOSED FRAMEWORK ARCHITECTURE

This section outlines the architecture of the proposed framework, featuring a DL-based multi-controller load balancing strategy and two main modules. The first module tackles security challenges arising from imbalanced data in SDN-CIoT environments and predicts future traffic loads. The second module is dedicated to routing optimization using DRL. Proposed framework is shown in a Fig 2.

A. DL-based Multi-Controller Load-Balancing Strategy

In our system model, the proposed framework introduces a novel DL-based multi controller load-balancing strategy aimed at reducing the computational load on the main controller for attack detection and traffic prediction, thereby enhancing overall system performance. The main controller acts as the framework's core, overseeing both the security and prediction controllers. A hierarchical communication model facilitates interaction between the main controller and these specialized controllers, utilizing OpenFlow for east/west bound interface communication among multiple controllers. Importantly, while the security and prediction controllers communicate with the main controller, they do not directly communicate with each other. The strategy employs a DL (CNN) model to assess average loads on controllers and identifies underutilized controllers based on CPU load, controller bandwidth status, and memory usage.

DL based multi-controller load-balancing scheme is shown in a Fig 3. This strategy is implemented in the main controller. It consists of four main components. For example load estimator component is used to measure the load of every controller. Controller adaption component select the target controller. Similarly, switch selection and switch migration components select and migrate the switches to the target controller respectively. There are some important assumptions while designing this strategy.

- In order to maintain the load-balancing, switches can be accessed and migrate from one controller to other controller.
- All controllers can not be overloaded at the same time.
- Only main controller has a global view of the entire network and it will decide the switch migration.

The algorithm 1 describes the proposed DL based load-balancing strategy.

B. First Module

The first DL-based module addresses two challenges: detecting attacks and anomalies from the imbalance data generated by SDN-CIoT, as well as predicting future traffic load in the SDN-CIoT network.

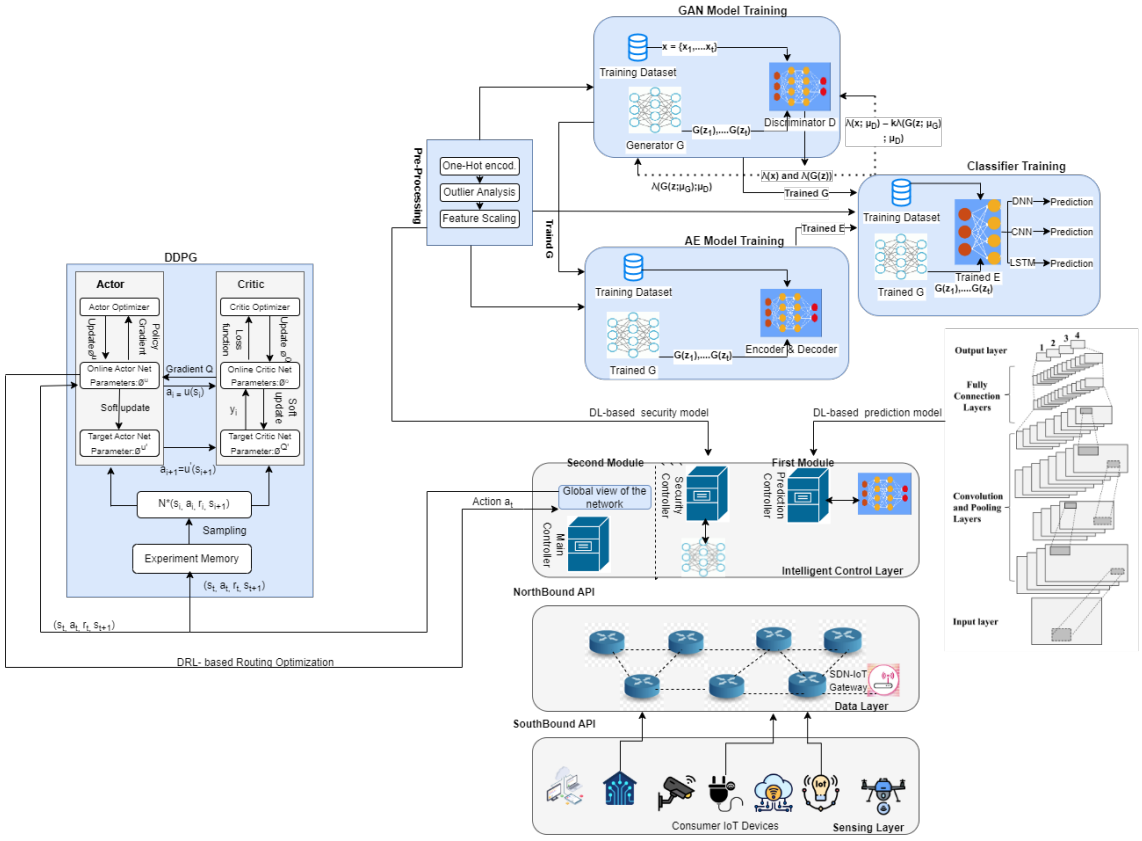


Fig. 2. The proposed framework's architecture comprises three layers: the sensing-actuator layer, the data layer, and the intelligent control layer. Within the third layer, there are three controllers: the main controller, the security controller, and the prediction controller.

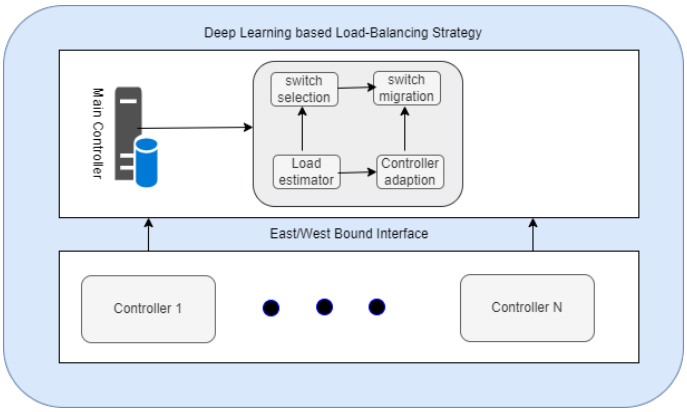


Fig. 3. Deep Learning Based Multi-Controller Load-Balancing Strategy.

Algorithm 1 DL based load-balancing strategy

1: **INPUT** : Controller Threshold
2: **Output** : Load-Balancing
3: If(controller load ¿ threshold)
4: Load estimator component is executed
5: Execute controller adaption module
6: For switch selection, third module is executed
7: Switch migration component shift the switches
8: Load status updation of both controllers
9: end if
10: **OUTPUT**: Load-balancing completed

1) Anomalies Detection in imbalance data in SDN-CIoT: To tackle the data imbalance challenge in the SDN-CIoT environment for precise and efficient minor class attack classification, we introduced a DL architecture based on Generative Adversarial Networks (GANs) for synthetic traffic generation. Specifically, we utilized an advanced version of GAN known as Bounded Equilibrium GAN. The process comprises four stages: 1) Pre-processing; 2) Train GAN; 3) Train AE; and 4) Train Prediction Model . Here, we describe all stages in detail. During the data processing, the raw data has undergone preparation and cleaning to make it compatible with a DL algorithm. Various methods have been employed to process the dataset. Initially, lines containing non-numeric characters or empty values are eliminated to minimize their influence on the test model's performance. Since DL algorithms excel with numerical data, non-numeric values are transformed into numerical equivalents using the label encoder, particularly sklearn. Additionally, to prevent unexpected impacts on model performance due to segment order, the output label is encoded just once. Data standardization is conducted through the Min-Max scalar function to enhance model performance.

After data-preprocessing, we construct and train the generative model with the synthetic data generation module.

Our chosen generative model is the Boundary Equilibrium Generative Adversarial Network, which operates akin to an AE. The generator is fashioned with the same architecture as the decoder of the discriminator, while the discriminator itself mirrors a symmetric five-layer AE model. Initially, the system classifies the provided dataset into distinct classes, subsequently creating generative models for each segmented sub-dataset. These models then contribute to training the generative model. Generative models are established in equal numbers as classes, with each dedicated to generating synthetic data corresponding to a specific class post-training. Determining the termination criterion for training holds pivotal importance in leveraging the generative model for detecting rare class attacks. This decision significantly impacts anomaly detection efficacy since it directly influences the synthetic data utilized for training the detection model. The generative model unique ability to gauge training convergence through the equilibrium concept sets apart it from other GAN models.

During third step, the AE model undergoes training to execute dimension reduction and feature extraction tasks, laying the foundation for an efficient anomaly detection model. Notably, the design of the generative model discriminator mirrors the architecture of the AE in our proposed model. Following the construction and training of an AE model on the expanded dataset, the trained encoder is utilized for the feature extraction process. Importantly, in the detection models, the trained encoder serves as the input layer, prioritized solely for feature extraction and restricted from further learning during detection model training.

At the last step of predictive model training, we employed Deep Neural Network (DNN), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) algorithms for anomaly categorization. The predictive models are classified into three groups for a thorough comparative analysis.

- Naive model which is LSTM
- CNN is combined with AE and is called an advanced DL model.
- $G-CNN_{AE}$, which is the proposed model.

2) Future Traffic Load Prediction in SDN-CIoT: The DL-based future traffic prediction sub-module comprises three stages: the data collection stage, the training stage, and the prediction stage. In the data collection stage, the prediction controller for traffic load prediction periodically gathers all information from switches. This controller maintains and records the traffic load sequence, denoted as $TrLo$, for every switch within the last Q time slots. Each time slot has a length represented by Δ. If we calculate the traffic load of an individual switch j in the previous time slot p, it is denoted as $trlo_p^j$. Subsequently, the vector $TrLo^j$ representing the previous traffic load of switch j, with a length of Q, is defined as follows: $TrLo^j = \{trlo_p^j, trlo_{p-1}^j, ..., trlo_{p-Q+1}^j\}$. Here, Q signifies the total number of considered previous time slots, a value influenced by two factors: 1) Input data complexity, and 2) Training performance. The prediction controller then proceeds to gather the traffic load data from all switches and organizes it into the form of a traffic load matrix, denoted as $TrLo = \{TrLo^1, TrLo^2, ..., TrLo^M\}$. After gathering data, the matrix $TrLo$ serves input for the dataset. In the upcoming time slot, the prediction controller registers the traffic volume in the form of actual future network flows, denoted as $trlo_{p+1} = \{trlo_{p+1}^1, trlo_{p+1}^2, ..., trlo_{p+1}^M\}$, which is used as the output for training dataset. Following a multitude of time slots, the prediction controller accumulates a vast collection of such dataset. These labeled real data are then employed for the training of deep CNN.

During the training Stage, we employ the Deep Convolutional Neural Network (D-CNN) model, which is trained on the traffic load matrix $TrLo$ generated in the previous stage. Studies have shown that D-CNN performs notably well when trained on matrix-based datasets [39]. Additionally, research from [40] compares training performance across various output formats, with results indicating that overly complex outputs can adversely affect training performance and accuracy. When a single D-CNN model is used for predicting future traffic load for all switches, the process becomes computationally expensive and relies on the complete $trlo_{p+1}$ as output, subsequently decreasing the accuracy of future traffic load predictions. To mitigate this, we employ N D-CNNs, each dedicated to a specific switch, calculating the traffic load prediction for its corresponding switch. This approach allows the prediction controller to use the network traffic volume of each switch as the output of its corresponding D-CNN. As an example, the Training dataset for $D-CNN^j$ is represented as $(x_{input}, y_{output}) = (TrLo, trlo_{p+1}^j)$.

At the prediction stage, the prediction controller engages in forecasting upcoming traffic volume and evaluates the accuracy of the predictions. During this stage, the weight matrix from each trained deep-CNN is utilized for predicting upcoming traffic volume. This process involves forward propagation as described earlier. All D-CNNs output is stored as a whole matrix of traffic volume $TrLoPr_{p+1} = \{trlopr_{p+1}^1, trlopr_{p+1}^2, ..., trlopr_{p+1}^M\}$. As described earlier, the actual forthcoming traffic volume at temporal interval $(p+1)$ is stored as $trlo_{p+1} = \{trlo_{p+1}^1, trlo_{p+1}^2, ..., trlo_{p+1}^M\}$. Hence, following equation is used for Prediction Accuracy (PA).

$$PA = \frac{1}{P \times M} \sum_{p=0}^{p-1} \sum_{j=1}^{M} \frac{|trlopr_{p+1}^j - trlo_{p+1}^j|}{trlo_{max}^j} \qquad (1)$$

In equation 1, P stands for the number of time slots, and $trlo_{max}^j$ signifies the highest traffic load observed for switch j. In the second module, which is utilized for routing optimization in SDN-IoT, the prediction of future traffic load is represented as $\kappa_s(t) = TrLoPr_{p+1}$.

C. Second Module(Agent Layer)

In the second module of our proposed framework, we implement DRL for routing optimization. Traditional routing algorithms assume decisions are based solely on current traffic loads, which works well under constant conditions but fails with irregular and complex network traffic patterns. To address this, our DRL module uses predicted traffic load data to optimize routing, aiming to prevent congestion and enhance

system performance in SDN-CIoT environments. The main controller gathers data on data imbalances, anomalies, and future traffic predictions, enabling the DRL agent to interact intelligently for optimized routing decisions. This approach improves throughput, reduces packet loss, and minimizes latency and jitter.

1) DRL based Routing Optimization in SDN-CIoT Environment:

- State: We consider routing as taking place within a unit of time, with each unit equivalent to one time step. Consequently, the total routing time between the source switch s_s and the destination switch s_d is denoted as $\mathbb{T}$. The DRL agent evaluates the reward for a transmission task within a single unit time slot, which encompasses determining the time required to select the subsequent SDN-enabled hop switch and transmit data to it. The DRL agent leverages four factors in the reward calculation: 1) Future traffic load prediction at switch s is $\kappa_s(t)$, 2) message holding rate of the flow table $\rho_s(t)$, 3) channel holding rate between the SDN controller and the switch $\sigma_s(t)$, and 4) message input frequency $\lambda_s(t)$. For a given switch node $s_i, i = (1, 2, 3, ...)$ during a unit time $t, t = (1, 2, 3, ...)$, these four factors collectively define the actual state of the system. Upon incorporating these four factors:

$$\begin{aligned} s(t) = \kappa_{s_1}(t), \kappa_{s_2}(t), \kappa_{s_3}(t), ...\kappa_{s_N}(t) \\ \rho_{s_1}(t), \rho_{s_2}(t), \rho_{s_3}(t), ...\rho_{s_N}(t) \\ \sigma_{s_1}(t), \sigma_{s_2}(t), \sigma_{s_3}(t), ...\sigma_{s_N}(t) \\ \lambda_{s_1}(t), \lambda_{s_2}(t), \lambda_{s_3}(t), ...\lambda_{s_N}(t) \end{aligned} \tag{2}$$

In equation 2, $\rho_{s_i} = \frac{\mathbb{F}s_i(t)}{\mathbb{F}s_i}$ signifies the capacity of the flow table in the SDN-enabled switch s_i. In this context, $\mathbb{F}s_i(t)$ represents the number of flow entries currently accommodated within switch s_i at any given time (t), while $\mathbb{F}s_i$ indicates the maximum number of flows that can be supported by switch s_i.

- Action: Smart and intelligent routing relies heavily on the selection of the next hop (switch). In the action stage, the primary responsibility of the DRL agent is to identify and choose the next available switch for data transfer. This action stage is depicted by Equation (6).

$$\mathbb{P}(t) = \mathbb{P}^{prest}_{s_1}(t), \mathbb{P}^{prest}_{s_2}(t), \mathbb{P}^{prest}_{s_3}(t), ...\mathbb{P}^{prest}_{s_N}(t) \tag{3}$$

As shown in a equation 3, $\mathbb{P}^{prest}_{s_i}(t)$ can be defined in the vector form $\mathbb{P}^{prest}_{s_i}(t) = \{\mathbb{P}^{prest}_{s_i,s_j}(t) | J \in \{1, 2, 3...N\}, J \neq I\}$. $\mathbb{P}^{prest}_{s_i,s_j}(t)$ shows the relation between the switch s_i and the switch s_j. Every element of $\mathbb{P}^{prest}_{s_i,s_j}(t) \in [0, 1]$, where $\mathbb{P}^{prest}_{s_i,s_j}(t) = 0$ means there is no connection between switch s_i and switch s_j at any unit time t and $\mathbb{P}^{prest}_{s_i,s_j}(t) \in [0, 1]$ shows the switch s_j weight that which is selected as next hop of switch s_i.

- Reward: In DRL, the efficiency and effectiveness of the agent's actions are assessed through the reward function. Consequently, the reward associated with each action varies. Within the proposed technique, key QoS parameters defining the reward function include throughput, latency, jitter, and packet loss rate. The reward function is articulated in equation 4.

$$RW(t) = \frac{1}{|Trans|} \sum_{i \in Trans} \beta RW^{QoS}_{s_i}(t) \tag{4}$$

Before delving into equation 4, it is important to acknowledge that the transmission task within any unit time t encompasses two distinct phases. In the first phase, SDN-enabled switch s_i transfers the data, and in the subsequent phase, the data is directed to the SDN-enabled destination switch s_j via a communication link. In equation 4, $|Trans|$ signifies the maximum number of data transmission jobs during any given unit time t. Parameter β is function tuning parameter, the values of which is adjusted to optimize the function, either by maximizing or minimizing it.

QoS rewards are defined in equation 5.

$$\begin{aligned} RW^{QoS}_{s_i}(t) = \sum_{j \in \{1,2,3,4...N\}, j \neq i} [\mathbb{P}^{prest}_{s_i,s_j}(t) + \\ Throughput_{s_i,s_j} - PLR_{s_i,s_j} - \\ Latency_{s_i,s_j} - Jitter_{s_i,s_j}] \end{aligned} \tag{5}$$

In equation 5, $Throughput_{s_i,s_j} - PLR_{s_i,s_j} - Latency_{s_i,s_j} - Jitter_{s_i,s_j}$ shows that throughput $Throughput_{s_i,s_j}$, packet loss rate PLR_{s_i,s_j}, $Latency_{s_i,s_j}$ and jitter $Jitter_{s_i,s_j}$ effect the QoS reward.

2) Deep Deterministic Policy Gradient : Deep Deterministic Policy Gradient (DDPG) is a reinforcement learning algorithm that merges deep learning with policy gradient methods to address problems with continuous action spaces. Unlike traditional methods such as Q-learning, which struggle with continuous actions requiring discretization, DDPG excels in such environments by using an actor-critic architecture. This setup involves two neural networks: the actor network accepts the current state and outputs continuous actions, approximating the optimal policy. Meanwhile, the critic network evaluates these actions by estimating the expected cumulative reward (Q-value) based on state-action pairs. This approach enhances learning efficiency and stability in environments where actions are continuous.

- DDPG: The fourth layer of our proposed model (DRL agent) utilizes DDPG, as illustrated in Figure 03. It employs the actor-critic model of DRL, wherein the actor comprises the actor network and the target actor network denoted as $\tau(s|\theta^{\tau})$ and $\tau^{'}$, respectively. Similarly, the critic includes the main critic network and the target critic network $\eta(s, a)$ and $\eta^{'}$, respectively. The structure of both the main network and the target network is the same. The current policy is determined by the actor-network

$\tau(s|\theta^\tau)$, which maps states to actions. The critic network $\eta(s,a)$ employs the Bellman equation for learning, and typically, the output of the actor serves as the input for the critic.

- Sample Collection: The exploration policy is employed to generate samples from the environment, and sample records $(s(t), a(t), r(t), s(t+1))$ are stored in a replay buffer B following the DDPG mechanism. Here, $s(t)$ and $a(t)$ denote the initial state and policy network output, respectively. Additionally, the action $a(t)$ is executed on the state $s(t)$, resulting in the corresponding rewards $r(t)$ and the subsequent state $s(t+1)$. The procedure for the sample collection process in SDN-CIoT is outlined in Algorithm 2.
- Training: The training process is represented in an equation (9)

$$Train(\theta) = \frac{1}{M}\sum_t (y(t) - \eta(s(t), a(t)|\theta^\eta))^2 \quad (6)$$

Deep Q-learning is used to train the critic net. As shown in equation 6, the actor network takes the state $s(t)$ as input and provides the action $a(t)$ as an output. Then, the critic network takes the action $a(t)$ as input and provides $\eta(s(t), a(t)|\theta^\eta)$ as an output.

$$\eta(s(t),a(t)) \longleftarrow \eta(s(t),a(t)) + \zeta(r(s(t),a(t))) + \omega_{a(t+1)}\eta(s(t+1),a(t+1)) - \eta(s(t),a(t)) \quad (7)$$

In equation 6, target Q-value $y(t)$ is defined as follow

$$y(t) = r(t) + \omega\eta^{'}(s(t+1), \tau^{'}(s(t+1)|\theta^{\tau^{'}})|\theta^{\eta^{'}}) \quad (8)$$

As shown in equation 8, the summation of reward and Q-value $r(t)+\omega\eta^{'}(s(t+1))$ gives the target value. The input state $s(t+1)$ gives the output action $\tau^{'}(s(t+1)|\theta^{\tau^{'}})$. By using the policy gradient technique, the gradient of the actor-network is given as:

$$\frac{\delta J(\theta^\tau)}{\delta\theta^\tau} = Z_s[\frac{\delta\eta(s,a|\theta^\eta)}{\delta a}\frac{\delta\tau^{'(s|\theta^\tau)}}{\delta\theta^\tau}] \quad (9)$$

Parameters updation process is explained in equation 10

$$\bigtriangledown_{\theta^\tau} J \approx \frac{1}{M}\sum_t \bigtriangledown_a \eta(s,a|\theta^\eta)|_{s=s(t),a=\tau(s(t))} \bigtriangledown_{\theta^\tau}\tau(s|\theta^\tau)|s(t) \quad (10)$$

Against the same state $s(t)$, main actor provides multiple actions. Hence, different actions can be used as an input for main critic to achieve different Q values. Equation 11 and equation 12 update the target network

$$\theta^{\eta^{'}} \longleftarrow \phi\theta^\eta + (1-\phi)\theta^{\eta^{'}} \quad (11)$$

$$\theta^{\tau^{'}} \longleftarrow \phi\theta^\tau + (1-\phi)\theta^{\tau^{'}} \quad (12)$$

Detailed training mechanism of the second module is elaborated in Algorithm 3.

Algorithm 2 Data sample collection process from the underlying environment

```
Initialization of buffer B
Initialization of both main critic and main actor networks η(s,a|θ^η), τ(s|θ^τ) along with their weights θ^η, θ^τ respectively.
Initialization of η' and τ' along with their weights θ^η' ⟵ θ^η and θ^τ' ⟵ θ^τ
for epic = one to Trans do
    state initialization s(t)
    for t=one to T do
        a(t) = τ(s(t)|θ^τ)
        a(t) and r(t)
        s(t+1)
        B (s(t), a(t), r(t), s(t+1))
    end for
end for
```

Algorithm 3 Training process of second module

```
for epic = 1 to Trans do
    for t=1 to T do
        Transition from buffer B (s(t), a(t), r(t), s(t+1))
        y(t) = r(t) + ωη'(s(t+1)τ'(s(t+1)|θ^τ')|θ^η')
        Train(θ) = (1/M) Σ_t (y(t) − η(s(t), a(t)|θ^η))^2
        (1/M) Σ_t ▽_a η(s,a|θ^η)|_{s=s(t),a=τ(s(t))} ▽_{θ^τ} τ(s|θ^τ)|s(t)
        θ^η' ⟵ φθ^η + (1−φ)θ^η' , θ^τ' ⟵ φθ^τ + (1−φ)θ^τ'
    end for
end for
```

V. Experiments and Evaluations

This section outlines the parameter tuning and implementation strategies for the framework modules. It then details the experimental setup and results, followed by a comparative analysis to assess the framework's effectiveness. The proposed modules are rigorously evaluated on an imbalance data to determine their significance in security and traffic prediction.

A. Parameters Tuning, implementation, and metrics of the Proposed Framework

1) First Module Implementation: The first module has two sub-modules: 1) Security sub-module and 2) Prediction sub-module.

- Security sub-module: This sub-module aims to detect and classify attacks and anomalies in imbalance data from diverse SDN-CIoT devices. The security sub-module utilizes a GAN discriminator and generator, each with three layers and 80 neurons in the hidden layer. The AE functions as a feature extractor with a similar architecture to the discriminator, and GAN training stops at a threshold of 0.058 or 280 epochs, while AE training stops at 300 epochs. For classification, DNN, CNN, and LSTM models, each with two hidden layers, are used. The DL algorithms are categorized into naive DL models

(DNN, CNN, LSTM), advanced DL models (DNN_{AE}, CNN_{AE}), and GAN-based DL models ($G-LSTM$, $G-DNN_{AE}$, $G-CNN_{AE}$). The models are compared using four metrics: Accuracy, Precision, Recall, and F1-score.

- Prediction sub-module: The proposed framework includes a specialized Prediction SDN Controller that uses a deep CNN model, advantageous for handling high-dimensional input layers. The deep CNN efficiently captures spatial and temporal data through its convolutional layers. These layers filter input data and pass the results to subsequent layers, while pooling layers reduce redundancy and aggregate neuron outputs, enhancing spatial feature extraction. Various filters in convolutional layers integrate their outcomes for fully connected layers, significantly reducing computational load and improving feature extraction efficiency. The convolution operation extracts distinctive input features using learnable filters, which include weights and biases. In this context, $F^{(l_1)}$ represents a filter, with the k^{th} filter is denoted as $F_k^{(l_1)}$ The resulting feature map from this operation is detailed below.

$$\begin{aligned} v_{i,j,k}^{(l_1)} &= (U^{(l_1-1)} * F_k^{(l_1)})(i,j) + \omega_{bk}^{(l_1)} \\ &= \sum_{p=1}^{P} \sum_{m=1}^{M'} \sum_{n=1}^{N'} \omega_{m,n,p} a_{i+m,j+n,p}^{(l-1)} + \omega_{bk}^{(l_1)} \end{aligned} \tag{13}$$

$$a_{i,j,k}^{(l_1)} = f(v_{i,j,k}^{(l_1)}) \tag{14}$$

2) Second Module Implementation: The second module of the proposed framework is a CNN model with two hidden layers using ReLU and Tanh activation functions. It trains a DDPG on a random number of nodes over 400 simulation epochs, calculating cumulative rewards in up to 25 steps per episode before updating the network. Efficient convergence is achieved with a learning rate of 0.1 and a discount factor of 0.9. Experiments show that the framework converges faster with these settings. Additionally, this module's routing optimization performance, measured by throughput and packet loss rate, is compared to the traditional OSPF protocol.

B. Experimental Setup

We constructed an SDN-CIoT environment using Mininet 2.3.0 for simulation. Deep Learning models integrated into ONOS SDN controllers using the TensorFlow framework in Python, specifically version 2.12.0. Simulations are conducted on a laptop equipped with an 8th generation Intel Core i9 processor, 16 GB of RAM, and a 1TB hard disk. The simulation environment includes switches and two types of devices: Non-Attacked Devices (NAD) and Attacked Devices (AD). Initially, each switch manages 50 devices, with an 85% to 15% ratio of NAD to AD at the start. To evaluate framework performance, we systematically reduce the proportion of AD. Switches are randomly positioned with distances varying between 10m and 20m. Testing focuses on imbalanced data generated by devices in the SDN-CIoT setup, with devices split in an 85% to 15% ratio. Wireless communication adheres to IEEE 802.11g standards, with each link operating at a 6 Mbps data rate. The simulation includes 5 to 70 switches, enabling diverse performance scenarios to be simulated effectively.

1) Security sub-module Performance: We evaluate the performance of the security sub-module and compare it with both the naive and advance deep learning models in terms of their attack detection accuracy using imbalance data from Non-Attacked and Attacked Devices within the SDN-CIoT environment. We conducted security module testing using two different types of imbalance data. In the first type, we had 85% normal traffic and 15% abnormal or attacked traffic. In the second type, we used 95% normal traffic and 5% attacked traffic. The comparison is presented in Fig 4 and Fig 5

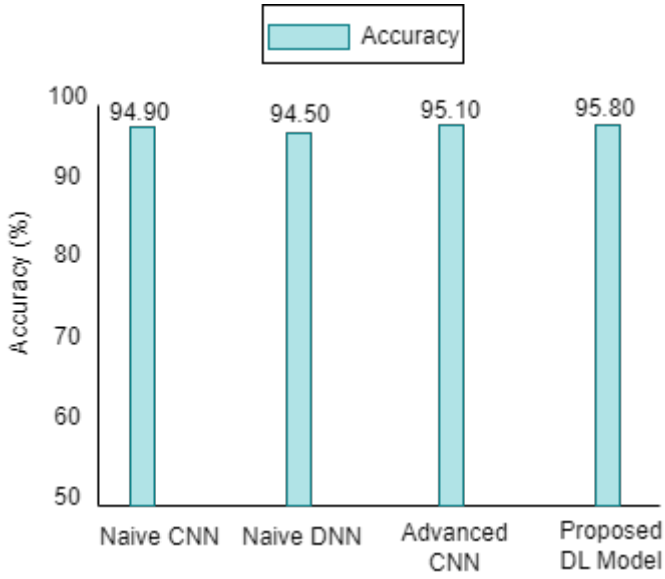


Fig. 4. A comparison is conducted among DL models, advance DL models, and the proposed GAN-based security sub-module. This comparison is based on the accuracy of attack detection in the SDN-CIoT environment when the ratio between Non-Attacked and Attacked devices is 85% to 15%, respectively. P.T denotes the Proposed Technique.

2) Future Traffic Prediction sub-module Performance: Secondly, we assessed the performance of the future traffic prediction sub-module and compared it with S-CTP and DTP, as previously done in [35]. We used prediction accuracy as the metric for comparison. Additionally, we kept the number of switches at 16 and the time slot length at 1 second. The time slot is a crucial parameter, and we evaluated the proposed sub-module under various time slot settings. After conducting extensive experiments, we observed that the accuracy of all three approaches increased as the time slot value increased. However, when the time slot value reached 50 seconds, the prediction accuracy plateaued. Consequently, we set 40 seconds as a threshold, indicating that the proposed deep learning-based prediction sub-module learned sufficiently from the training data. Results are shown in a Fig 6.

3) Routing Optimization module Performance: After performing imbalance data attack detection and future traffic prediction in the SDN-CIoT environment, we then executed

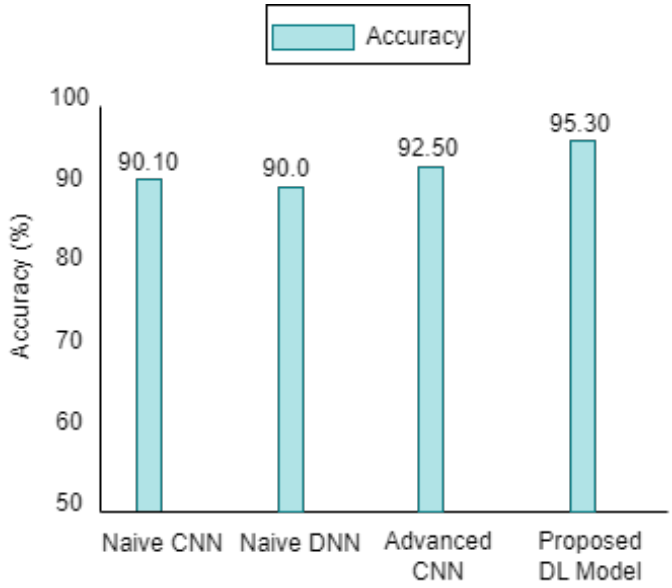


Fig. 5. A comparison is conducted among naive DL models, advance DL models, and the proposed GAN-based security sub-module. This comparison is based on the accuracy of attack detection in the SDN-CIoT environment when the ratio between Non-Attacked and Attacked devices is 95% to 5%, respectively. P.T denotes the Proposed Technique.

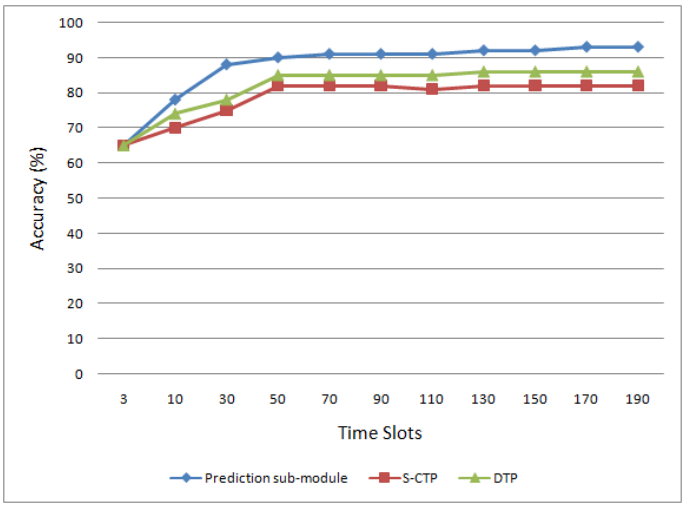


Fig. 6. A comparison is conducted among the prediction sub-module, S-CTP, and DTP based on prediction accuracy.

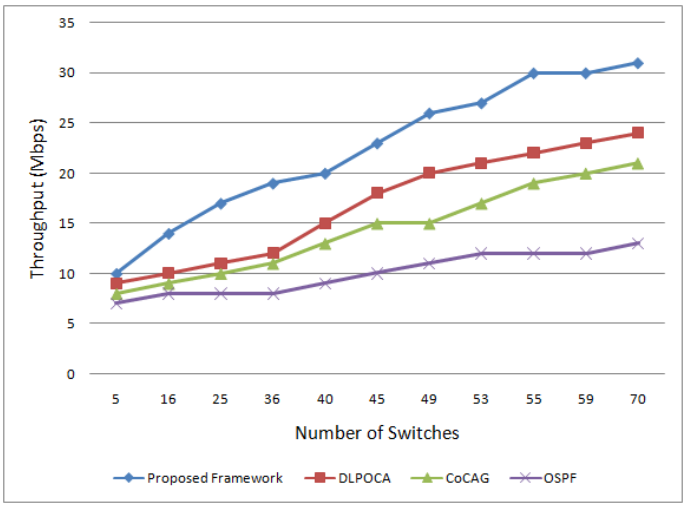


Fig. 7. A comparison is conducted among the proposed framework, DLPOCA, CoCAG, and OSPF based on throughput.

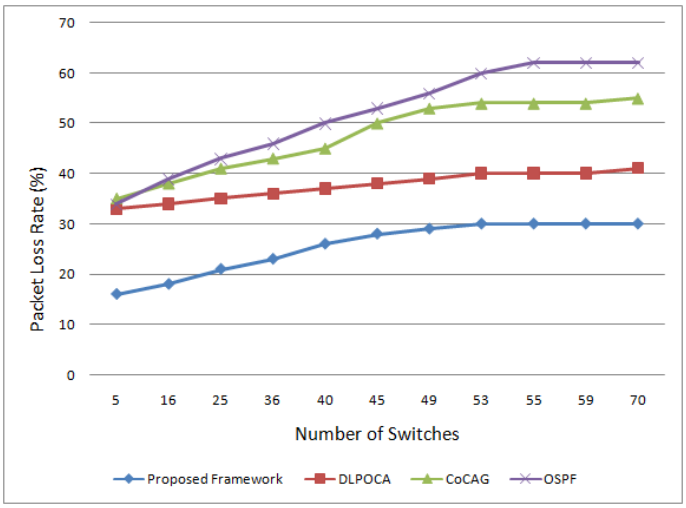


Fig. 8. A comparison is conducted among the proposed framework, DLPOCA, CoCAG, and OSPF based on packet loss rate.

routing optimization module to achieve optimized routing. This module leverages the results from the first module to perform secure and intelligent routing optimization. We employed two key metrics: throughput and packet loss rate for evaluation. Furthermore, we conducted a comparison of proposed framework with the latest approaches as well as traditional routing protocols to assess its efficiency and efficacy. The results are shown in a Fig 7 and Fig 8.

C. Datasets Description

1) SDN-IoT Dataset: According to [41], no real-world dataset exists which includes IoT within an SDN environment, a scenario distinct from standard non-IoT cases [42]. To address this, the authors [41] generated their own dataset by simulating and integrating IoT devices within an SDN framework, producing two versions of the SDN-IoT dataset as CSV files. The only difference between the datasets is the volume of sensors used in the testbed, with one dataset intended for training and the other for testing purposes. The dataset includes both normal and malicious traffic, collected through 20 experiments of 30-35 minutes each. The first 15 experiments featured IoT and background traffic, while the last 5 focused solely on IoT traffic. Malicious traffic generated using five types of attacks: DoS, DDoS, port scanning, OS fingerprinting, and fuzzing. An SDN controller processed the traffic and generated 33 features, resulting in 125,973 training records and 22,543 testing records across 6 classes. The original datasets, initially containing 27.9 million and 30.2 million records, reduced to meet specific requirements. The

data distribution of the SDN-IoT dataset is presented in the Table II.

D. Conducting Experiments with the SDN-IoT Dataset

In this subsection, we tested and evaluated the proposed framework on four key metrics accuracy, precision, recall, and F1-score for imbalance data attack detection and classification. It's important to note that the SDN-IoT dataset is a imbalance datasets. Therefore, the proposed framework achieves maximum performance on this dataset. The results of the naive, advanced, and proposed DL model for attack detection and classification are presented in Table III.

VI. Results and Discussion

This scientific study employed an AI-based framework to address imbalance data security attacks, prediction of future traffic load, and routing optimization in an SDN-CIoT environment. Extensive experimentation reveals that the security sub-module achieved an exceptional accuracy of 95.8% in detecting and classifying minor class attacks in imbalance data, when the ratio of non-attacked to attacked devices is 85% to 15%, respectively. This performance surpasses both naive and advanced DL models. The success of the security sub-module on imbalance data can be attributed to the GAN, which generates plausible synthetic data for minor attack traffic and effectively addresses the data imbalance problem. The detailed results are presented in Fig 4. Additionally, the accuracy of the security sub-module on imbalance data, with a non-attacked to attacked device ratio of 95% to 5%, is depicted in Fig 5. The performance of the prediction sub-module is assessed by comparing it with state-of-the-art approaches such as S-CTP and DTP, using prediction accuracy as the metric for comparison. Fig 6 shows that the prediction sub-module achieved higher prediction accuracy compared to both approaches. The second module of the proposed framework based on DRL is employed for routing optimization. In experiments, we have assessed routing optimization based on two key metrics: throughput and packet loss rate. While simulating routing optimization, we compared four approaches: 1) The AI-based Framework, 2) DLPOCA, and, 3) CoCAG, and 4) OSPF. The results indicate that the AI-based framework outperformed the other three approaches in terms of throughput and packet loss rate. The excellent performance of the AI-based framework is attributed to its routing based on the secure and future traffic load prediction sub-modules. First, these two sub-modules operate to provide attack-free and predicted future traffic loads to the second module, which then performs routing. Next, the AI-based framework without the security sub-module also outperforms OSPF. Even though it lacks a security module, it excels in routing based on future traffic load prediction. The results of these four approaches are presented in Fig 7 and Fig 8. Finally, we tested the security sub-module on a publicly available imbalance dataset (SDN-IoT) using four key metrics: accuracy, precision, recall, and F1-score. The findings are displayed in Table III. It is clear from the table that the proposed security sub-module achieved higher accuracy and F1-score values—94.5% and 91.6% respectively—on extremely minor classes such as Fuzzing, which has a 0.037% weight in the overall population.

Despite the numerous advantages of the AI-based framework, it is important to acknowledge that there are associated disadvantages. This framework relies on computationally expensive models, demanding substantial computational resources to fulfill its tasks. Furthermore, implementing it in a real-world environment can result in significant costs. In summary, the benefits of the AI-based proposed framework far outweigh its drawbacks, making it a viable solution for real-world applications.

VII. Conclusion

In this article, we proposed an SDN and AI based framework for Consumer Internet of Things network to protect from imbalance data attacks, predict future traffic load, and perform routing optimization. Specifically, SDN architecture is integrated with CIoT network to handle its distributed architecture and heterogeneous consumer electronic devices. Then, an AI model based on GAN and CNN deployed at SDN control plane to enhance minor class attacks detection and future traffic load prediction mechanism respectively. We proved the effectiveness of the proposed framework in terms of accuracy, throughput and packet loss rate through experimental evaluation on the imbalance SDN-IoT dataset. We also compared the performance of the proposed framework against some recent state-of-the art technique.

Acknowledgment

This work was supported by the "Merging technology for Object Detection in Unmanned Ariel Vehicles" project under grant number (113-2222-E-155-006- to Q.M.U) funded by the National Science and Technology Council (NSTC), R.O.C.(Taiwan)

Funding

This project was funded by Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah under grant No. (RG-6-611-43), the authors, therefore, acknowledge with thanks DSR technical and financial support.

References

[1] R. Kumar, D. Javeed, A. Aljuhani, A. Jolfaei, P. Kumar and A. K. M. N. Islam, "Blockchain-Based Authentication and Explainable AI for Securing Consumer IoT Applications," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 1145-1154, Feb. 2024, doi: 10.1109/TCE.2023.3320157.

[2] C. K. Wu, C. -T. Cheng, Y. Uwate, G. Chen, S. Mumtaz and K. F. Tsang, "State-of-the-Art and Research Opportunities for Next-Generation Consumer Electronics," in IEEE Transactions on Consumer Electronics, vol. 69, no. 4, pp. 937-948, Nov. 2023, doi: 10.1109/TCE.2022.3232478.

[3] C. Park, J. Lee, Y. Kim, J. -G. Park, H. Kim and D. Hong, "An Enhanced AI-Based Network Intrusion Detection System Using Generative Adversarial Networks," in IEEE Internet of Things Journal, vol. 10, no. 3, pp. 2330-2345, 1 Feb.1, 2023, doi: 10.1109/JIOT.2022.3211346.

[4] J. Su et al., "Trustworthy IAP: An Intelligent Applications Profiler to Investigate Vulnerabilities of Consumer Electronic Devices," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 4605-4616, Feb. 2024, doi: 10.1109/TCE.2023.3347651.

TABLE II
SDN-IOT IMBALANCE DATASET DISTRIBUTION

Class	Training	Weight%	Testing	Weight%
Normal	135,000	96.25%	24,300	94.67%
DoS	3,000	2.14%	800	3.11%
DDoS	1,500	1.07 %	400	1.56%
Port scanning	600	0.43%	92	0.35%
OS fingerprinting	100	0.071%	45	0.18%
Fuzzing	52	0.037%	30	0.11%
Total	**140,252**	**100%**	**25,667**	**100%**

TABLE III
COMPARISON OF THREE MODEL TYPES: NAIVE DL MODEL, ADVANCE DL MODEL, AND THE PROPOSED FRAMEWORK, FOR CLASSIFICATION OF NORMAL AND ABNORMAL NETWORK TRAFFIC ON AN IMBALANCE SDN-IOT DATASET.

		←——Normal——→			←——Abnormal——→		
Classifier	**A**	**R**	**P**	**F1**	**R**	**P**	**F1**
DNN	82.4 %	97.3 %	69.3 %	80.5 %	68.3 %	96.5 %	81.0 %
CNN	83.5 %	98.0 %	70.4 %	81.3 %	70.5 %	96.9 %	81.5 %
$LSTM$	84.6 %	98.1 %	72.0 %	83.2 %	71.0 %	97.4 %	82.1 %
DNN_{AE}	86.2 %	99.0 %	79.0 %	88.4 %	73.0 %	98.7 %	84.5 %
CNN_{AE}	87.0 %	99.0 %	80.0 %	88.8 %	75.0 %	98.3 %	85.6 %
$G.LSTM$	89.5 %	98.7 %	78.5 %	88.3 %	73.5 %	98.7 %	84.2 %
$G.DNN_{AE}$	92.3 %	98.4 %	85.7 %	91.3 %	82.3 %	98.0 %	91.3 %
$G.CNN_{AE}$	94.5 %	98.2 %	86.3 %	91.9 %	85.4 %	97.0 %	91.6 %

A = Accuracy, R = Recall, P= Precision, F1= F1- score, $G.LSTM, G.DNN_{AE}, G.CNN_{AE}$ =Proposed Framework

[5] P. Chanak and I. Banerjee, "Congestion Free Routing Mechanism for IoT-Enabled Wireless Sensor Networks for Smart Healthcare Applications," in IEEE Transactions on Consumer Electronics, vol. 66, no. 3, pp. 223-232, Aug. 2020, doi: 10.1109/TCE.2020.2987433.

[6] W. Xia, Y. Wen, C. H. Foh, D. Niyato and H. Xie, "A Survey on Software-Defined Networking," in IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 27-51, Firstquarter 2015, doi: 10.1109/COMST.2014.2330903.

[7] Ngangbam Indrason, Goutam Saha, Exploring Blockchain-driven security in SDN-based IoT networks, Journal of Network and Computer Applications, Volume 224, 2024, 103838, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2024.103838.

[8] D. Javeed, M. S. Saeed, I. Ahmad, P. Kumar, A. Jolfaei and M. Tahir, "An Intelligent Intrusion Detection System for Smart Consumer Electronics Network," in IEEE Transactions on Consumer Electronics, vol. 69, no. 4, pp. 906-913, Nov. 2023, doi: 10.1109/TCE.2023.3277856.

[9] J. K. Samriya, C. Chakraborty, A. Sharma, M. Kumar and S. K. Ramakuri, "Adversarial ML-Based Secured Cloud Architecture for Consumer Internet of Things of Smart Healthcare," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 2058-2065, Feb. 2024, doi: 10.1109/TCE.2023.3341696.

[10] D. -J. Kim, N. G. B. Amma and V. Sarveshwaran, "A Novel Split Learning-Based Consumer Electronics Network Traffic Anomaly Detection Framework for Smart City Environment," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 4197-4204, Feb. 2024, doi: 10.1109/TCE.2024.3367330.

[11] Ikhlas Al-hammadi, Mingchu Li, Sardar M.N. Islam, Esmail Al-Mosharea, Collaborative computation offloading for scheduling emergency tasks in SDN-based mobile edge computing networks, Computer Networks, Volume 238, 2024, 110101, ISSN 1389-1286, https://doi.org/10.1016/j.comnet.2023.110101.

[12] K. Janani and S. Ramamoorthy, "Countermeasure sdn-based iot threats using blockchain multicontroller," International Journal of High Performance Systems Architecture, vol. 11, no. 3, pp. 117–128, 2023, https://doi.org/10.1504/IJHPSA.2023.130190.

[13] A. Montazerolghaem, "Efficient Resource Allocation for Multimedia Streaming in Software-Defined Internet of Vehicles," in IEEE Transactions on Intelligent Transportation Systems, vol. 24, no. 12, pp. 14718-14731, Dec. 2023, doi: 10.1109/TITS.2023.3303404.

[14] X. Guo, H. Lin, Z. Li and M. Peng, "Deep-Reinforcement-Learning-Based QoS-Aware Secure Routing for SDN-IoT," in IEEE Internet of Things Journal, vol. 7, no. 7, pp. 6242-6251, July 2020, doi: 10.1109/JIOT.2019.2960033.

[15] R. Chaganti, W. Suliman, V. Ravi, and A. Dua, "Deep learning approach for sdn-enabled intrusion detection system in iot networks," Information, vol. 14, no. 1, p. 41, 2023, https://doi.org/10.3390/info14010041.

[16] P. S. Manocha and R. Kumar, "Improved spider monkey optimization-based multi-objective software-defined networking routing with block chain technology for internet of things security," Concurrency and Computation: Practice and Experience, vol. 34, no. 11, p. e6861, 2022.

[17] G. Kumar, N. Shamanth et al., "A comprehensive research on deep learning based routing optimization algorithms in software defined networks," in 2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT). IEEE, 2023, pp. 1–5.

[18] S. I. Popoola, A. L. Imoize, M. Hammoudeh, B. Adebisi, O. Jogunola and A. M. Aibinu, "Federated Deep Learning for Intrusion Detection in Consumer-Centric Internet of Things," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 1610-1622, Feb. 2024, doi: 10.1109/TCE.2023.3347170.

[19] M. A. Razib, D. Javeed, M. T. Khan, R. Alkanhel and M. S. A. Muthanna, "Cyber Threats Detection in Smart Environments Using SDN-Enabled DNN-LSTM Hybrid Framework," in IEEE Access, vol. 10, pp. 53015-53026, 2022, doi: 10.1109/ACCESS.2022.3172304.

[20] M. Ibrar, L. Wang, G. -M. Muntean, J. Chen, N. Shah and A. Akbar, "IHSF: An Intelligent Solution for Improved Performance of Reliable and Time-Sensitive Flows in Hybrid SDN-Based FC IoT Systems," in IEEE Internet of Things Journal, vol. 8, no. 5, pp. 3130-3142, 1 March1, 2021, doi: 10.1109/JIOT.2020.3024560.

[21] A. Montazerolghaem, "Software-Defined Internet of Multimedia Things: Energy-Efficient and Load-Balanced Resource Management," in IEEE Internet of Things Journal, vol. 9, no. 3, pp. 2432-2442, 1 Feb.1, 2022, doi: 10.1109/JIOT.2021.3095237.

[22] J. Ahmed, H. H. Gharakheili, C. Russell and V. Sivaraman, "Automatic Detection of DGA-Enabled Malware Using SDN and Traffic Behavioral Modeling," in IEEE Transactions on Network Science and Engineering, vol. 9, no. 4, pp. 2922-2939, 1 July-Aug. 2022, doi: 10.1109/TNSE.2022.3173591.

[23] S. Lee, J. Kim, S. Shin, P. Porras and V. Yegneswaran, "Athena: A Framework for Scalable Anomaly Detection in Software-Defined Networks," 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, USA, 2017, pp. 249-260, doi: 10.1109/DSN.2017.42.

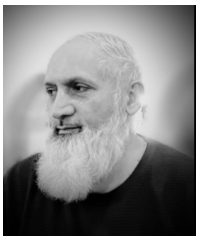
[24] Zabeehullah, F. Arif, N. A. Khan, Q. Mazhar ul Haq, M. Asim and S. Ahmad, "An SDN-AI-Based Approach for Detecting Anomalies in Imbalance Data within a Network of Smart Medical Devices," in IEEE Consumer Electronics Magazine, doi: 10.1109/MCE.2024.3389292.

[25] I. Ahmed, G. Jeon and A. Ahmad, "Deep Learning-Based Intrusion Detection System for Internet of Vehicles," in IEEE Consumer

Electronics Magazine, vol. 12, no. 1, pp. 117-123, 1 Jan. 2023, doi: 10.1109/MCE.2021.3139170.
[26] M. Yamauchi, Y. Ohsita, M. Murata, K. Ueda and Y. Kato, "Anomaly Detection in Smart Home Operation From User Behaviors and Home Conditions," in IEEE Transactions on Consumer Electronics, vol. 66, no. 2, pp. 183-192, May 2020, doi: 10.1109/TCE.2020.2981636.
[27] C. -C. Chuang, Y. -J. Yu and A. -C. Pang, "Flow-Aware Routing and Forwarding for SDN Scalability in Wireless Data Centers," in IEEE Transactions on Network and Service Management, vol. 15, no. 4, pp. 1676-1691, Dec. 2018, doi: 10.1109/TNSM.2018.2865166.
[28] A. R. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta and M. Alikarar, "A smart home energy management system using IoT and big data analytics approach," in IEEE Transactions on Consumer Electronics, vol. 63, no. 4, pp. 426-434, November 2017, doi: 10.1109/TCE.2017.015014.
[29] W. Zhong, N. Yu and C. Ai, "Applying big data based deep learning system to intrusion detection," in Big Data Mining and Analytics, vol. 3, no. 3, pp. 181-195, Sept. 2020, doi: 10.26599/BDMA.2020.9020003.
[30] Ahmed Hazim Alhilali, Ahmadreza Montazerolghaem, Artificial intelligence based load balancing in SDN: A comprehensive survey, Internet of Things, Volume 22, 2023, 100814, ISSN 2542-6605, https://doi.org/10.1016/j.iot.2023.100814.
[31] S. S. Tripathy, S. Bebortta, M. I. u. Haque, Y. Zhu and T. R. Gadekallu, "Toward Multi-Modal Deep Learning-Assisted Task Offloading for Consumer Electronic Devices Over an IoT-Fog Architecture," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 1656-1663, Feb. 2024, doi: 10.1109/TCE.2024.3365107.
[32] J. Gao et al., "Omni SCADA Intrusion Detection Using Deep Learning Algorithms," in IEEE Internet of Things Journal, vol. 8, no. 2, pp. 951-961, 15 Jan.15, 2021, doi: 10.1109/JIOT.2020.3009180.
[33] G. Wang, Y. Zhao, J. Huang and W. Wang, "The Controller Placement Problem in Software Defined Networking: A Survey," in IEEE Network, vol. 31, no. 5, pp. 21-27, 2017, doi: 10.1109/MNET.2017.1600182.
[34] F. Tang, B. Mao, Z. M. Fadlullah and N. Kato, "On a Novel Deep Learning-Based Intelligent Partially Overlapping Channel Assignment in SDN-IoT," in IEEE Communications Magazine, vol. 56, no. 9, pp. 80-86, Sept. 2018, doi: 10.1109/MCOM.2018.1701227.
[35] F. Tang, Z. M. Fadlullah, B. Mao and N. Kato, "An Intelligent Traffic Load Prediction-Based Adaptive Channel Assignment Algorithm in SDN-IoT: A Deep Learning Approach," in IEEE Internet of Things Journal, vol. 5, no. 6, pp. 5141-5154, Dec. 2018, doi: 10.1109/JIOT.2018.2838574.
[36] Zabeehullah et al., "DQQS: Deep Reinforcement Learning-Based Technique for Enhancing Security and Performance in SDN-IoT Environments," in IEEE Access, vol. 12, pp. 60568-60587, 2024, doi: 10.1109/ACCESS.2024.3392279.
[37] M. Aslam, D. Ye, A. Tariq, M. Asad, M. Hanif, D. Ndzi, S. A.Chelloug, M. A. Elaziz, M. A. Al-Qaness, and S. F. Jilani, "Adaptive machine learning based distributed denial-of-services attacks detection and mitigation system for sdn-enabled iot," Sensors, vol. 22, no. 7, p. 2697, 2022, https://doi.org/10.3390/s22072697.
[38] A. Montazerolghaem and M. H. Yaghmaee, "Load-Balanced and QoS-Aware Software-Defined Internet of Things," in IEEE Internet of Things Journal, vol. 7, no. 4, pp. 3323-3337, April 2020, doi: 10.1109/JIOT.2020.2967081.
[39] F. Tang et al., "On Removing Routing Protocol from Future Wireless Networks: A Real-time Deep Learning Approach for Intelligent Traffic Control," in IEEE Wireless Communications, vol. 25, no. 1, pp. 154-160, February 2018, doi: 10.1109/MWC.2017.1700244.
[40] N. Kato et al., "The Deep Learning Vision for Heterogeneous Network Traffic Control: Proposal, Challenges, and Future Perspective," in IEEE Wireless Communications, vol. 24, no. 3, pp. 146-153, June 2017, doi: 10.1109/MWC.2016.1600317WC.
[41] A. Kaan Sarica and P. Angin, "A Novel SDN Dataset for Intrusion Detection in IoT Networks," 2020 16th International Conference on Network and Service Management (CNSM), Izmir, Turkey, 2020, pp. 1-5, doi: 10.23919/CNSM50824.2020.9269042.
[42] F. De Keersmaeker, Y. Cao, G. K. Ndonda and R. Sadre, "A Survey of Public IoT Datasets for Network Security Research," in IEEE Communications Surveys & Tutorials, vol. 25, no. 3, pp. 1808-1840, thirdquarter 2023, doi: 10.1109/COMST.2023.3288942.

Zabeehullah received the B.Sc. Software Engineering degree from the Department of Software Engineering, University of Engineering and Technology, Taxila, Pakistan, in 2012, and the M.S. (Computer Science) degree from the Department of Computer Science, National University of Computer and Emerging Sciences, Islamabad, Pakistan in 2015. Currently, he is pursing Ph.D. degree from the Department of Computer Software Engineering National University of Science and Technology, Islamabad, Pakistan. His research interests include Internet of Things (IoT), Internet of Medical Things (IoMT), Software Defined Network, SDN-IoT, AI, Deep Learning applications in SDN-IoT for Quality of Service, routing communication, and security.

Qazi Mazhar ul Haq is currently serving as an Assistant Professor in the Department of International Bachelor Program in Informatics and Computer Science and Engineering at Yuan Ze University, Taiwan. He has held visiting professor positions at the Harbin Institute of Technology, China, and was previously a faculty member at the National University of Sciences and Technology, Pakistan. He actively participates in various IEEE committees, including the IEEE Consumer Technology Society, IEEE Biometrics Council, IEEE Sensors Council, and IEEE Systems Council. Additionally, he has contributed as a guest editor for a special issue in ACM Transactions on Asian and Low-Resource Language Information Processing. Dr. Haq has also been a Postdoctoral Research Fellow at the National Taipei University of Technology, Taiwan. He earned his PhD in Electronics and Computer Engineering from the National Taiwan University of Science and Technology. His research focuses on unmanned ariel vehicles, autonomous vehicles, Networking, object detection, incremental learning, anomaly detection, image processing, deep learning, and 3D object detection. Contact him at qazi@saturn.yzu.edu.tw.



Fahim Arif received his Ph.D. in Software Engineering from National University of Sciences and Technology (2009), however he completed his research work in Carleton University, Ottawa, Canada (2007). He attended professional development program at George Mason University, Fairfax, USA (2013). He is Senior Member IEEE. His research interests include software engineering, SQA, Remote sensing and machine learning application in software domain.

Nauman Ali Khan received his Ph.D. degree in Communication and Information Systems at the University of Science and Technology of China, Hefei, China. He is currently serving as an Assistant Professor at Department of Computer Software Engineering National University of Science and Technology, Islamabad, Pakistan. His research interests include Machine Learning, Social Network Analysis, Wireless Big Data Mining, Social-Tie Inference, and Prediction.

Muhammad Shahid Anwar is currently working as an Assistant Professor in the Department of AI and Software at Gachon University, Seongnam, South Korea. He received his Ph.D. degree in Information and Communication Engineering from the School of Information and Electronics, Beijing Institute of Technology, Beijing, China in 2021. His research interests include 360-degree videos, Metaverse, and healthcare systems.

Wadee Alhalabi received his master and PhD. Degree in electrical and computer engineering from the University of Miami in 2004 and 2008 respectively in Machine Learning. He is a professor of artificial intelligence in the computer science department KAU. He published about 100 articles in the area on virtual reality, machine learning.