Classification and Characterization of Encoded Traffic in SCADA Network using Hybrid Deep Learning Scheme

Love Allen Chijioke Ahakonye, Gabriel Chukwunonso Amaizu, Cosmas Ifeanyi Nwakanma, Jae Min Lee, and Dong-Seong Kim

***Abstract*—The domain name system (DNS) has evolved into an essential component of network communications, as well as a critical component of critical industrial systems (CIS) and Supervisory Control and Data Acquisition (SCADA) network connection. DNS over HTTPS (DoH) encapsulating DNS within hypertext transfer protocol secure (HTTPS) does not eliminate network access exploitation. This paper proposes a hybrid deep learning model for the early classification of encoded network traffic into one of the two classes: DoH and NonDoH. They can be malicious, benign, or zero-day attacks. The proposed scheme incorporates the swiftness of the convolutional neural network (CNN) in extracting useful information and the ease of long short-term memory (LSTM) in learning long-term dependencies. The simulation results showed that the proposed approach accurately classifies the encoded network traffic as DoH or NonDoH and characterizes the traffic as benign, zero-day, or malicious. The proposed robust hybrid deep learning model had high accuracy and precision of 99.28%, recall of 99.75%, and AUC of 0.9975 at a minimal training and testing time of 745s and 0.000324 s, respectively. In addition to outperforming other compared contemporary algorithms and existing techniques, the proposed technique significantly detects all attack types. This study also investigated the impact of the SMOTE technique as a tool for data balancing. To further validate the reliability of the proposed scheme, an industrial control system SCADA (ICS-SCADA) dataset, in addition to two (2) other cyber-security datasets (NSL-KDD and CICDS2017), were evaluated. Mathews correlation coefficient (MCC) was employed to validate the model performance, confirming the applicability of the proposed model in a critical industrial system such as SCADA.**

***Index Terms*—CIS, DNS, encoded network traffic, hybrid deep learning, IIoT, IoT, network intrusion, SCADA security**

I. Introduction

Manuscript received June 21, 2022; revised August 10, 2022; approved for publication by Lee, Heejo Division 3 Editor, December 9, 2023.

This research work was partly supported by the Priority Research Centers Program through NRF funded by MEST (2018R1A6A1A03024003 (34%)). partly supported by the Ministry of Science and ICT (MSIT), Korea, under the Innovative Human Resource Development for Local Intellectualization support program (IITP-2024-2020-0-01612 (33%)) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation), and partly supported by the Ministry of Science and ICT (MSIT), Korea (1711175292/2022-IT-RD-0084-01 (33%)).

The authors are with the department of IT convergence engineering, Kumoh National Institue of Technology, Gyeongsangbuk-do, South Korea, email: {loveahakonye, cosmas.ifeanyi, ljmpaul, dskim}@kumoh.ac.kr, gabriel4amaizu@gmail.com.

D.-S. Kim is the corresponding author.

Digital Object Identifier: 10.23919/JCN.2023.000067.

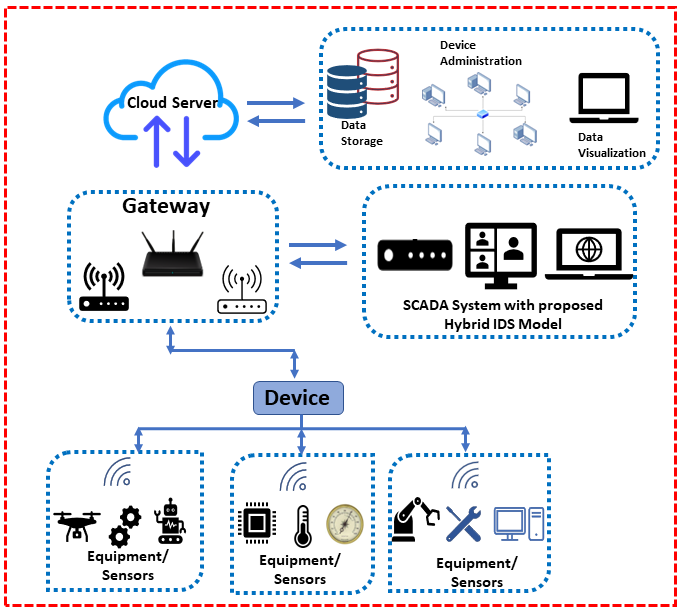

Fig. 1. IIoT flow diagram with the proposed SCADA network hybrid DL model.

THE conventional information security approaches usually do not offer comprehensive protection to critical industrial systems like supervisory control and data acquisition (SCADA). It is due to the prevailing exploitation mechanisms, approaches, and the constant emergence of new attacks. Researchers and security professionals constantly study various intrusion detection techniques with neglect of systems and situations under intrusion [1]–[4]. Hence, only information security is insufficient in addressing overall security issues in critical systems such as the SCADA network. There is a need to consider network communication, and traffic protocols [5]–[9].

The SCADA network presents surveillance of disseminated industrial facilities, control, and procedures for real-time information production in the industrial Internet of things (IIoT). CIS smart factory employs SCADA systems to automate industrial processes such as manufacturing and power generation for real-time service delivery [10], [11]. Regardless of the significance, the SCADA network is not adequately secured and hence vulnerable to a series of attacks that can be detrimental to time-critical operations [12]. The point of

1229-2470/24/$10.00 © 2024 KICS

TABLE I
TABLE OF ABBREVIATIONS.

Abbreviation	Meaning
AUC	Area under curve
B_Norm	Batch normalization
B5G	Beyond 5G
CIS	Critical industrial systems
CNN	Convolutional neural networks
DDoS	Distributed denial of service
DL	Deep learning
DoH	DNS over HTTPS
DoS	Denial of service
DNS	Domain name service
ECN	Ethernet consist network
FTP	File transfer protocol
HMI	Human machine interface
HTTPS	Hypertext transfer protocol secured
IETF	Internet Engineering Task Force
IDS	Intrusion detection system
ICS	Industrial control system
IIoT	Industrial Internet of things
IoT	Internet of things
IP	Internet protocol
LSTM	Long short-term memory
MiTM	Man in the middle
ML	Machine learning
NIDS	Network intrusion detection system
NIST	National Institute of Standards and Technology
PCC	Pearson's correlation coefficient
PLCs	Programmable logic units
ReLU	Rectified linear unit
RFE	Recursive feature elimination
R2L	Remote to local
RTUs	Remote terminal units
SCADA	Supervisory control and data acquisition
SSH	Secure shell
U2R	User to remote
XGBoost	Extreme gradient boosting

these attacks is predominantly devices and connections such as network communication and transmission protocols, application servers, sensors, and actuators. Securing the SCADA network transmission requires protecting the communication and transmission protocols, hence the need for further protection of the domain name service (DNS). Despite the soaring intricacy of SCADA systems and their increased susceptibility to the internet, the security of the network and communication protocols may not apply a structured approach. Due to prevailing security threats, bypass of security measures, data egress, and incompatibility issues. Therefore, an in-depth hybrid deep learning protection based on a real-time apprehensive approach is necessary for securing the SCADA network communication system [12]. Fig. 1 represents a typical network flow of the IIoT showing the placement of the proposed hybrid model.

The rise of the beyond 5G ($B5G$) and IIoT has prepared the way for more device connectivity, thus enhancing the utilization of internet and network communication protocols [2], [7] and increasing its susceptibility to vulnerability and attacks. The security of these interconnections is consequently critical, particularly in the context of the SCADA network. The DNS, which works like a phonebook for users, has been a major facilitator in Internet communication. With the sole function of translating domain names to Internet protocol (IP) addresses. There may be some floating questions about the preparedness of DNS for $B5G$ and IIoT. Nevertheless, there is a certainty that DNS is essential [13]–[20].

DNS encoding involves enclosing data transmission between the server and a client using the DNS protocol [21]. The data is inside a standard DNS query, and the server may or may not give some encrypted data in the DNS feedback conversation. DNS encoding in SCADA network communication is due to difficulty in detection and prevention due to recursive hops traversed by data packets before the name server destination. Firewalls also fail to inspect the frequency and substance of DNS packets accurately. As a result, the SCADA network is vulnerable. The goal is to protect the communication protocol from cases of DNS data manipulation and hijacking, eavesdropping, and increasing the security of the communications in transit. However, the focus on the DoH is from the point of protecting only communications in transit to reduce vulnerability to attacks like man-in-the-middle (MiTM).

With the introduction of IP version 6 (IPv6), it is now possible to address a whole range of devices, something that was not possible with the almost exhausted IP version 4 (IPv4) [22]. Although the DNS function may seem unnoticeable, a failure in the network can hamper users' accessibility to SCADA resources via the internet [23]. DNS failures often result from attacks by an adversary, a trend that keeps increasing by the day [24]–[27]. This situation necessitates an efficient hybrid deep learning-based protection mechanism to keep up with the evolving attack trends. The National Institute of Standards and Technology (NIST) published a document outlining rules for safely deploying DNS to avoid the security risks associated with it [28]. The Internet Engineering Task Force (IETF) introduced the DNS over hypertext transfer protocol secured to improve DNS security (DoH). This new protocol helped to improve DNS security and privacy. However, this scheme is insufficient to mitigate against advancing vulnerability and attacks in critical systems like the SCADA network.

Attempts at SCADA network security protection have implemented firewalls and network intrusion detection systems; to subsist the challenges of attacks and vulnerabilities. A limited number of these studies focused on communication protocols, notably DNS. This study focuses on DNS protocols owing to their applicability in network traffic communications using HTTP/HTTPS. It is understandable as myriads of the network traffic communications generated by SCADA intrusions use DNS [25], [29]. To mitigate against various DNS attacks, a real-time, proactive method; capable of intelligently detecting and classifying the characteristics of SCADA network communication traffic is crucial. Given the time-critical operations of the SCADA system, reliability, low computation cost, and significant detection accuracy determine the performance of an efficient intrusion detection technique. The lack of a comprehensive technique that meets these demands remained an issue.

Hence, an intelligent hybrid deep learning technique for the swift detection and characterization of SCADA network traffic. Following the degradation of model performance with an increase in the feature-dimensionality [30], the novelty of the proposed model is proffering low computational cost and

enhanced detection accuracy by the employed capability of independent feature engineering technique. This model can be applied in any system with vulnerability issues, making it a good fit for real-time systems such as SCADA. Thereby mitigation against obfuscation by evading established IPs and ports. Labeling the input as the attack in the testing stage is not required in the presented scheme. Hence this study aimed to develop a model that intelligently predicts the character of the transit network traffic (if incoming traffic is non-DOH, benign-DoH, or malicious DoH). So, it can characterize transit communication traffic and eliminate the non-DoH and malicious DoH traffic, allowing only the benign DoH traffic and thereby improving the system's security.

This study uses an intelligent hybrid deep neural network to detect and characterize SCADA network communication traffic accurately. Furthermore, such an efficient ML scheme should include alternative metrics, such as the Mathews correlation coefficient (MCC) [31], for testing the proposed approach's reliability in real-world scenarios with unbalanced data. The main contributions of this paper are as below:

- To achieve efficient, real-time detection/classification, this research proposes an intelligent, time-efficient hybrid scheme that improves network security. The proposed classification model of the IDS in SCADA network communication emulates real-time traffic monitoring.
- This research illustrates the effect of the combination of convolutional neural network (CNN) and long short-term memory (LSTM) to learn long-term dependencies and the significant improvement of the classification.
- This study demonstrates the improvement of the swiftness, stability, and robustness of CNN by imploring batch normalization.
- The study also highlights the benefit of implementing dropout layers to tackle the issue of overfitting, which predominantly affects deep learning models.
- Also is the comparison of the performance of other models and the proposed architecture for precision, throughput, computational time, recall, and F1-score.
- The Mathews correlation coefficient validated the proposed models' reliability, in addition to the evaluation of the industrial control systems (ICS-SCADA) dataset.

The next Section II introduces the related studies on current and trending approaches for intrusion detection, while Section III thoroughly discusses the system methodology of the proposed deep hybrid neural network. In Section IV, the performance evaluation is presented and lastly, the conclusion is summarized in Section V. Table I is a list of major abbreviations used in this work.

II. Theoretical Background and Related Works

A. What is SCADA

SCADA systems are industrial automation networks that use software and hardware to acquire, analyze, and provide more precise data generated from sensors to control and monitor industrial operations [32]. These technologies are critical for the industrial environment to comprehend and achieve their processes for data-driven decisions to optimize operations. The software and hardware of a SCADA system components communicate in synchrony [32]. SCADA software analyzes and interprets hardware data configured for management and anomaly detection. The hardware consists of relays, sensors, and switches, with the principal role of acquiring critical operating data. This data goes to the programmable logic controllers (PLCs) or remote terminal units (RTUs), which convert it into an industry-standard protocol that can be processed and used for operational efficiencies [32]. The data is then transferred to the human machine interfaces (HMIs) for renderings of the processes via indicators, statistical reports, spreadsheets, alert signals, and patterns, analyzed for informed data-driven decisions. Before the widespread adoption of automation technology in the smart factory, industrial processes were controlled and managed manually. Organizations must control and monitor equipment and operations on a much larger scale and across longer distances as factories and processes get more extensive and more complex. The introduction of PLCs and RTUs in the industrial sector paved the way for developing SCADA systems.

Some basic features of the SCADA system are as follows:

1) *Acquisition of High-Performance Data:* A SCADA system should be capable of mission-critical swift data gathering via a database capable of logging data at high speed.
2) *Functionality:* A sound SCADA system provides a robust framework for process automation, triggering operations to complete actions and engaging users in predefined procedures.
3) *Ubiquitous Connectivity:* A high-quality SCADA system with ubiquitous connectivity allows any data in the system for connectivity to and from any location, enabling IoT-ready operations. It should support some actual implementations of message queuing telemetry transport (MQTT), simple network management protocol (SNMP), Internet of things (IoT), databases, and web services, that allow the aggregation of any data using communication techniques.

B. Characteristics and Attack Patterns of a SCADA Network

SCADA networks become more stable over time, as network applications do not join or leave regularly. Conventional networks often offer a wide range of protocols, such as HTTP, instant messaging, and voice over IP, whereas SCADA networks provide services like monitoring and controlling industrial processes and automation. Primarily, due to the polling mechanism utilized to collect data, most SCADA traffic is projected to be generated regularly. As a result, traffic patterns are not reliant on human activity, as is the case in traditional IoT networks [33]. Extensive research into traditional networks revealed that SCADA networks are very different from regular networks [33], [34]. The IEC-60870-5-104 (IEC-104) protocol is widely used in SCADA networks to manage sensitive facilities like power plants [34]. As the SCADA security significance grows, researchers are studying the characterization and modeling of SCADA traffic to develop

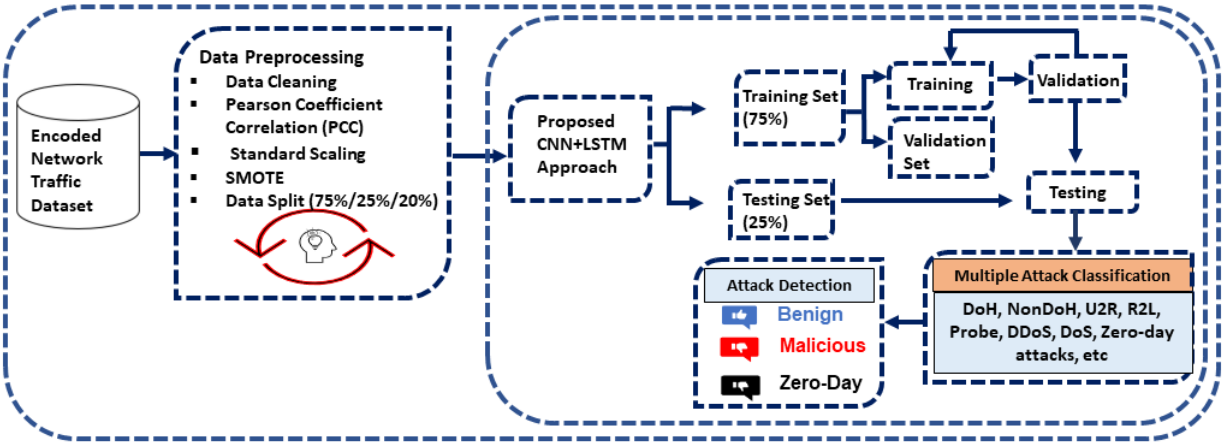

Fig. 2. Schematic representation of system model.

defense mechanisms based on the regularity of the polling mechanism used in SCADA systems, particularly the characterization of traffic caused by non-polling mechanisms such as spontaneous events. Authors [34] proposed a method based on the probabilistic suffix tree (PST) to find the underlying temporal patterns of spontaneous events, providing an insight into how the traffic flows between SCADA components varies over time. Providing evidence of the existence of patterns in data distinct from standard conventional network traffic patterns.

C. Related Works

A Series of research on intrusion detection with machine and deep learning (ML/DL) techniques with an emphasis on IoT; is limited to the emerging SCADA and smart factories. However, this study with insight into SCADA network intrusion detection utilizing ML and DL is required to ascertain the extent of work in the target domain. Ensemble learning approaches combine many classifiers to generate predictions that increase performance for attacks and protocols in IoT networks. Although this method produced improved outcomes, the approach is rather cumbersome due to a lack of processing speed [35]–[37].

Research has to detect malicious DNS activity in the DoH environment utilizing HTTP traffic [38]–[41]. These ML attempts investigated various mechanisms for anomaly detection/classification of DNS over HTTPS. However, it is brief with a lack of detailed scrutiny of robust approaches for a high-dimensional dataset. Similarly, [29] described a two-level strategy of using classifiers to recognize and categorize DNS over hypertext traffic. According to the authors, the study's key feature is its capacity to detect and classify DoH traffic with a limited amount of input data. As a result, the model is unsuited for Smart manufacturing activities due to its lack of robustness. In another study, [42] presented a hybridization strategy with a universal optimization methodology for detecting distributed denial of service (DDoS) attacks in IoT. A prototype version of the CICIDS2017 dataset investigated this method. The authors plan to test the model on distributed IDS because, while efficient, it lacked computational speed and was not robust enough for a smart factory.

The deep learning framework of CNN and LSTM networks or other combinations has significantly impacted computer vision as well as the intrusion detection [42]–[47]. It can comprehend the concept of high dimensionality, as well as execute feature extraction, classification, and data integration in heterogeneous IIoT datasets. Two fundamental aspects are hierarchical feature representations and a long-term understanding of dependencies in large-scale sequence data.

There have been attempts at the design of SCADA IDS. The paper by [43] evaluated the popular NSL-KDD dataset and offered a strategy for identifying attacks using LSTM and CNN. Despite the model's good performance, the categorization speed could be faster. [48] in a study presented the development of a SCADA system testbed to analyze the effects of attacks. Their approach investigated the KNN, random forest, naive Bayes, and decision tree classifiers. Another of their papers [49] conducted a flow-based intrusion detection study for a SCADA system using deep artificial neural networks. The proposed approach assessed both online and offline attacks. The strategy performed admirably but needs to evaluate more number of attacks.

[50] developed a technique for IDS in power grids that combines recursive feature elimination – extreme gradient boosting (RFEXGBoost) centered on feature selection with a majority-vote ensemble approach. When evaluated on publicly available datasets collected on a modest power grid testbed, achieved considerable detection rate, accuracy, recall, and precision. A technique for IDS based on a few-shot learning approach for attack detection in a SCADA network by [51] showed ambitious performance with few examples in identifying attacks. However, these current studies lacked time efficiency, which is vital in Smart factory operations.

As a result, this work provides a hybrid approach with time efficiency to harness the influence of cyber-security vulnerability and attack detection. The approach models an intelligent detection and characterization of the network traffic in a SCADA system.

While there have been studies into detecting and categoriz-

ing fraudulent DNS via HTTPS traffic, the problem here is that the research in the AI domain is not yet mature enough. Furthermore, there appear to be a small number of studies that utilize the entire dataset or provide some observations of the dataset used, and network parameters with readability, and are beneficial to the research community. Furthermore, AI techniques for detecting and classifying DNS over HTTP traffic assaults should have high accuracy and precision, recall, the area under the curve (AUC), minimal processing time, and reliability. In summary, unlike most others, this research uses the concept of combining two separate neural networks to solve the mentioned challenge of encapsulated DNS attacks.

Recently, various studies on DL approaches widely acknowledged for their resilience and ability to learn and predict important attributes from network traffic to resolve attacks on intrusion detection demonstrated improved performance [42]–[47]. However, these studies seem limited due to the focus of some models on attack classification while others performed detection. It is also that these models lack comprehensive intrusion detection of varying attack types and zero-day attacks. The main argument of this study is to address the salient requirement for security in IIoT, particularly SCADA networks, by using more precise techniques to attack detection and categorization of all attack types, including zero-day attacks. The proposed approach, in addition to enhanced model performance, seeks to reduce computational time effectively.

D. Summary Research Gaps from Related Works

Table IIsummarizes related works based on the identified limitation of existing studies leveraging the large, imbalanced, and high-dimensional CIRA-CIC-DoHBrw-2020 dataset. This table shows that most published works are difficult to reproduce due to insufficient details and transparent methodologies. In this work, a careful effort to ensure the reproducibility of the results is critical to growing the yet to be matured use of AI for SCADA vulnerability research. Also, Table III summarizes related works based on the combination of CNN and LSTM, highlighting the limitations of existing studies. Two limitations are evident. First is the limited SCADA datasets available such that most authors resorted to using available datasets such as NSL-KDD or KDD99. Secondly, in the case of the SCADA datasets, most authors' performance evaluation is limited to one or two datasets. In this work, we have adopted the use of repeated evaluation and MCC metrics to show the reliability of our proposed scheme to provide good accuracy at least computation time on four publicly available datasets. In addition, we have ensured the model is reproducible by adopting the principle of explainable AI.

TABLE II
SUMMARY OF RELATED WORKS BASED ON ENCODED CIRA-CIC-DOHBRW-2020 TRAFFIC DATASET (YES: $\surd$, NO: χ).

Study	**Used all dataset samples**	**Provided details of the neural network features**	**Ease of results reproducbility**
Detecting malicious DNS [38]	χ	χ	χ
Machine learning for DNS tunneling [39]	χ	χ	χ
Attack classification [40]	χ	$\surd$	χ
DoH traffic classification [41]	χ	$\surd$	χ
Detection of DoH tunnels [21]	$\surd$	χ	$\surd$
Proposed scheme (DNS traffic classification/attack detection)	$\surd$	$\surd$	$\surd$
Dataset	CIRA-CIC-DoHBrw-2020		

TABLE III
SUMMARY OF RELATED WORKS BASED ON CNN-LSTM APPROACH (YES: $\surd$, NO: χ, N/A: NOT AVAILABLE).

Author	**Year**	**Dataset**	**Improved detection accuracy**	**Low computational cost**
[42]	2020	CICIDS2017	$\surd$	χ
[43]	2020	NSL-KDD	$\surd$	$\surd$: Transformed standardized data into image form
[44]	2020	KDD99	$\surd$	N/A
[45]	2020	CICIDS2017	$\surd$	N/A
Proposed scheme	2022	CIRA-CIC-DoHBrw-2020, ICS-SCADA, NSL-KDD, CICIDS2017	$\surd$: In classifying and characterization of all attack types across all evaluated datasets	$\surd$: In classifying and characterization of all attack types across all evaluated datasets

III. METHODOLOGY

A. Description of the Power System SCADA Network

SCADA networks enable an ideal method for remote control and monitoring industrial resources. It is extensive in various industrial applications such as factory automation, water treatment, oil gas pipeline control, monitoring, power systems, and increased efficiency. It collects data from various production units and processes it accordingly. The programmable logic controllers in remote locations continuously monitor the unit components and relay that information to the central system. It increases efficiency by maintaining a manageable range of operational factors [53]. Fig. 3 is the power system SCADA network configuration diagram used to generate the dataset [52]. It comprises various parts, the first of which are power generators G1 and G2. R1 to R4 are intelligent electronic devices (IEDs) that control the breakers (on or off). The breakers are BR1 to BR4. In addition, are two lines, line one connects breaker one (BR1) to breaker two (BR2), and line two connects breaker three (BR3) to breaker four (BR4). Each IED is programmed to control one breaker. R1 controls BR1, and R2 controls BR2, respectively. Since they lack internal validation, IEDs use a distance protection technique that trips the breaker on detecting anomalies regardless of whether they are valid or contrived. The components and configuration of the

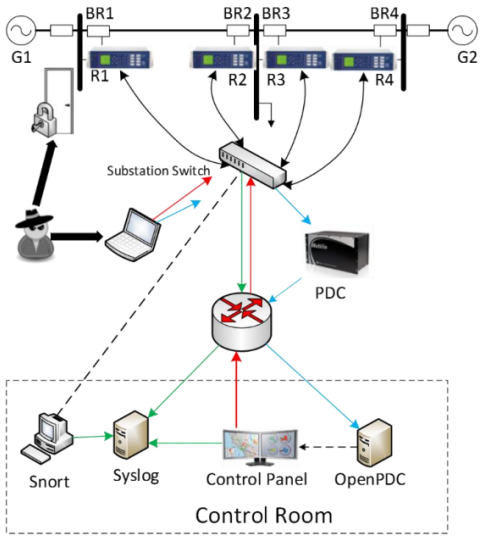

Fig. 3. Configuration of the power system scada network used to generate the SCADA dataset [52].

power system SCADA network define its uniqueness in terms of data generated, susceptibility, and attack methods [33], [34]. Therefore the features of the SCADA traffic pattern and characteristics contrast the ordinary internet traffic. Hence, it requires a robust approach for classifying and characterizing its encoded traffic. As a result, a practical, time-exigent IDS is necessary. The data includes twenty-nine (29) types of measurements from different phasor measurements (PMU). A phasor measurement unit (PMU) is a device that measures electrical waves on a power grid while synchronizing with an expected time source. This network comprises four PMUs that each measure 29 features for a total of 116 PMU measurement columns in the dataset. Each column's index is in the form "R#-Signal Reference," indicating the type of measurement from a PMU designated by "R#." The dataset contains 128 features. For more details of features, see the dataset description.

B. Attack Traffic and Types

The constant increase in the number of attack scenarios, combined with new and more complicated network and software configurations, necessitates the inclusion of real-time network traffic in data sets. This phenomenon has also led to the non-existence of a perfect network-based dataset [54]. To fully evaluate IDS approaches, more than one dataset is essential to avoid over-fitting to a specific dataset, limit the influence of false artifacts in a specific dataset, and analyze their techniques in a more global definition. IDS datasets contain a variety of attack scenarios. This attribute identifies the presence of zero-day, benign, and malicious network traffic in a dataset and returns true if the dataset includes any of the attributes. Additional information describing the specific attack types and their criteria are below:

1) *Zero-day Traffic:* This focuses on a zero-day vulnerability. A zero-day vulnerability is an identified attack yet to be addressed in the network traffic.
2) *Benign Traffic:* This is typical network traffic devoid of any form of intrusion or attack.
3) *Malicious Traffic:* In this scenario, intruders exploit the network traffic and bombard the target system with various attack types such as denial of service (DoS) and, DDos, MiTM, infiltration. It takes advantage of the functionality of the system's vulnerability to overwhelm the target.

C. System Methodology

To address the problems discussed in Section II, this work proposed the use of an intelligent CNN-LSTM for a time-efficient vulnerability and attack detection in a smart factory SCADA network. See Fig. 2 for system model architecture.

1) LSTM: LSTM is known for the capability of training long-term reliance on data. The main feature of using LSTM is the elimination of feature engineering [55]. LSTM network comprises collective units; these are three main elements, input port, forget port, and output port. They guide the upgrade, improvement, and excision of data enclosed in the unit. For generating the current state, the input gate uses a sigmoid function to govern the input data [56].

$$i_t = \varsigma[w_i.\,(f_{t-1}, f_t)^R + b_i], \tag{1}$$

where b_i and w_i denote offset and the weight matrix of the input gate. Furthermore, the input gate uses the tanh function to build a data vector for the current state. The proposed network determines the hidden state $\tilde{c_t}$ as thus: the use of the outcome of both the input and forget gates, the proposed network determines the hidden state $\tilde{c_t}$ as:

$$\tilde{c_t} = tanh[w_c.\,(f_{t-1}, f_t)^R + b_c], \tag{2}$$

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c_t}. \tag{3}$$

The forget gate then uses a sigmoid gate ς to delete unnecessary data from the input layer output f_t and preceding cell output f_{t-1}. Finally, multiply the data to combine it. The forget gate's output f_t looks like this:

$$f_t = \varsigma[w_f.\,(f_{t-1}, f_t)^R + b_f], \tag{4}$$

where b_f and w_f are offsetting the forget gate and weight matrix.

Lastly, the output gate selects useful characteristics based on the current cell state, the primary cell's result, and the new data. This output gate function, o_t, is denoted thus:

$$o_t = \varsigma[w_o.\,(f_{t-1}, f_t)^R + b_o]. \tag{5}$$

The resultant LSTM layer: $f_{outcome}$ is denoted by:

$$f_{outcome} = o_t \times tanh\,(c_t)\,. \tag{6}$$

2) CNN: CNN is an extensively utilized neural network with applications presently mainly in computer vision for image recognition, capable of efficiently extracting useful batches of data in a large sample. For this study, see CNN as expanded in Section III-D.

D. The Hybrid Framework

This section presents the proposed hybridization of the CNN-LSTM model; that opposes the computational time and accuracy of existing intrusion detection in IIoT (SCADA). Despite a series of researches demonstrating that the growth of growing network size and execution time results in an excellent rapid performance. The execution time and limitation in variable quantity are also essential concerns in IIoT (SCADA) IDS. Therefore, to resolve the determined limitations, this study considers multi-variate convolution, passing over interconnectivity to employ optimal parametric pooling and attain competent learning performance at a minimal execution time. Fig. 4 illustrates that the suggested approach is built with a standard convolution layer to generate completely detached variables and an LSTM layer for SCADA network traffic classification and characterization.

The input layer is the pre-processed min-max data normalization of the size $\mathbf{Q} \in \mathrm{X}^{\text{packsize x 14 x 1}}$. Apt to enhance the convergency of the input dataset is the batch normalization layer to guard against the overfitting of the model. The input data in this study normalized into tiers for each of the 30 mini-pack sizes. The procedure for this standard setting is in two phases: Normalization, resizing, and equaling. This procedure maintains the model learning process, impressively lowering the number of iterations needed for training deep learning. The batch-normalized data goes through (1x3) convolution layers with 12 kernels. In the convolutional layer, the 1D convolution operation of the kernel and input map is the sum of the dot product at a specific spatial coordinate (*x,y*) as follows:

$$\mathbf{Conv}_{\mathrm{a,\ b}} = \sum_{i} \mathrm{x}_j \mathrm{y}_j, \tag{7}$$

where x_j indicates the convolution kernel weight and y_j refers to the network traffic worth of the input dataset. The scalar bias input *q* controlled the convolution outcome and reckoned an established cost.

$$\mathbf{z}_{a,b} = \mathrm{Conv}_{a,b} + \mathrm{q}. \tag{8}$$

The output feature P is $\mathrm{X}^{\text{packsizex14x16}}$. The value 16 is the number of employed mechanisms.

The feature map is from a non-linear activation function *h* as the convolutional layer outcome. The rectified linear unit (ReLU) activation function is employed to minimize concerns of over-fitting. The ReLU generates a zero value for any value less than zero and forwards the value more than or equals zero. The process of the ReLU is as follows:

$$\mathbf{h}\ (\mathbf{P}) = \begin{cases} \mathbf{P} & \text{if P} \geq 0, \\ 0 & \text{if P} < 0. \end{cases} \tag{9}$$

Three different stream interconnectivity strategies are to compute the ReLU layer feature map result. The integration layer combines the processing units of the first and second streams, where the final feature maps integrate with the prior unit product using the new connection. For details, see Fig. 4. Moreover, this technique, known as substructuring, provides increased learning ability with the least execution time. The result feature map of the convolution, processed with the non-symmetric convolution kernel in the first stream, this aids in the extraction of subsurface parameters and improves model precision. A spatial composition layer then merges the resulting output of the non-symmetric convolution layers.

A concurrent CNN framework was to utilize asymmetric procedure with a sizeable architecture and attain low computational complexity. This framework is to obtain fast convergence of the training operation. However, a residual interconnection evades the issue of vanishing gradients and enhanced accuracy. The network architecture comprises three (3) correspondent procedures with various convolution intensities for feature extraction. The first procedure edges two convolutions (1x3) and (1x1) aggregated consecutively and passed into the max-pooling layer. The second procedure is three (3) convolution layers, the last two layers organized concurrently. The third procedure employed one max-pooling to minimize feature dimensions and obtain heterogeneous features. Each stage is combined and passed to an average pooling with size two (2) to decrease the execution time.

The average pooling layer output is input to the LSTM layer, then forwarded to the dense layer. The resulting LSTM structure goes to the dense layer. Following the dense layer is the data evaluation by the softmax and the fully connected layer for the classification and characterization of encoded SCADA network traffic [57]. Given the effective method for capturing long-term correlation structure by the LSTM, the proposed approach employs an LSTM layer following the average-pool layer. It consists of memory cells called neurons. The cells are the input, forget, and output gates, and to evaluate the input variables, each of the gates offers various functions. For instance, based on the cell status, the forget gate determines eliminating irrelevant information. To begin, the forget gate uses a sigmoid gate ς to delete redundant information from the additive layer result from f_t and prior cell result f_{t-1}. Lastly, multiplying, the gathered information is multiplied.

See Fig. 4 for a representation of the proposed model framework and Table IV for the complete network architecture of the proposed model, showing the features and specifications. Both classification and characterization of network traffic employed similar network architecture and parameters. It is to ascertain the viability of the proposed model for a real scenario in network traffic classification and detection. The ideal feature parameters setting of the proposed model used for the two layers (classification and detection) is in Table V. The proposed model performed optimally with the following training structure: extraction of most useful data features enabled by the adam optimizer with cross-entropy loss function, mini-batch size of 30 for 80 iterations, an initial learning rate of 0.001, ReLU activation function, and k-fold for cross-validation is 2.

E. Dataset and Data Pre-processing

This study evaluated four (4) publicly available datasets—the CIRA-CIC-DoHBrw-2020, NSL-KDD, and CICIDS2017 datasets from the Canadian Institute for Cybersecurity repository [58]. They were generating actual

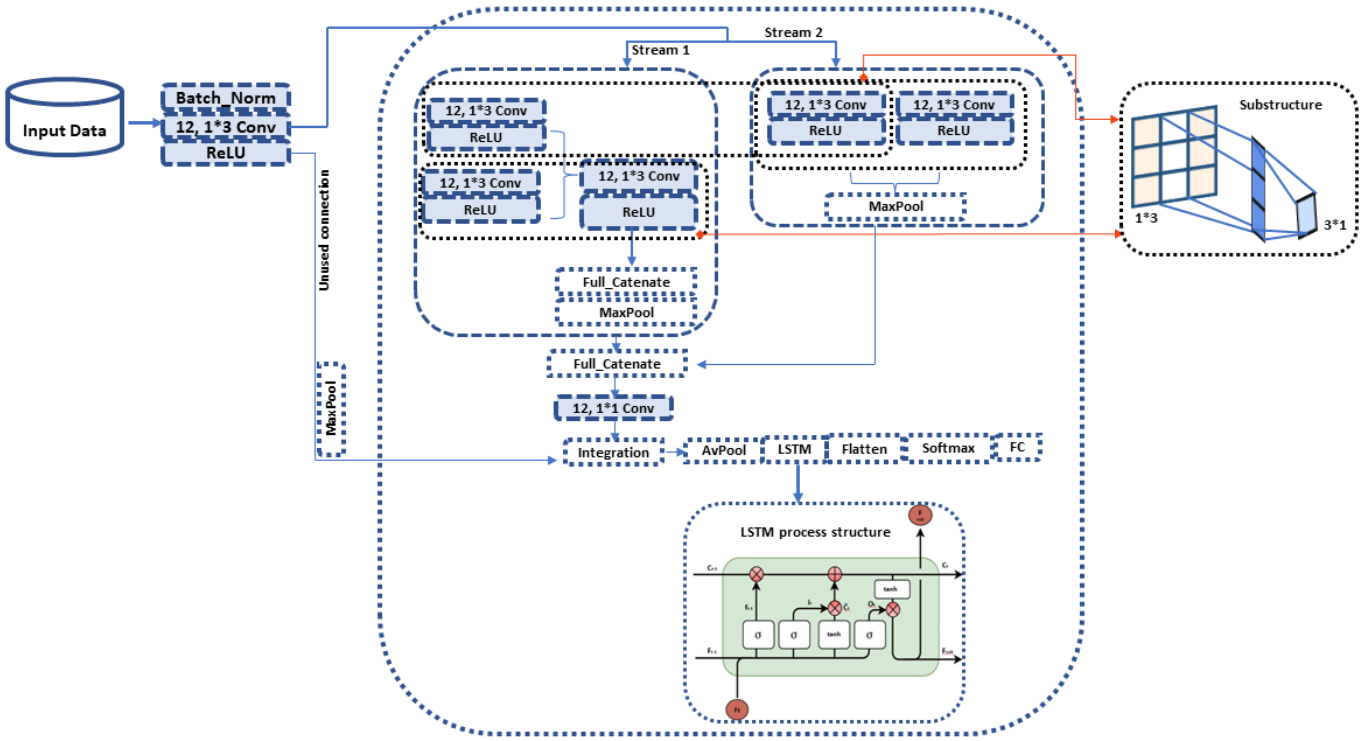

Fig. 4. The proposed hybrid model framework with fully connected layers.

TABLE IV
NETWORK ARCHITECTURE OF PROPOSED CNN+LSTM MODEL.

Module	Features	Specification
Input	[batchsize, 14, 1]	
Batch normalization	[batchsize, 14, 1]	
Conv1D_1	[batchsize, 14, 16]	12, 1x3, ReLU, padding
Conv1D_2	[batchsize, 14, 16]	12, 1x3, ReLU, padding
Conv1D_3	[batchsize, 14, 30]	28, 1x1, ReLU, padding
MaxPooling1D_1	[batchsize, 7, 30]	Pooling size=2
Conv1D_4	[batchsize, 14, 30]	28, 1x3, ReLU, padding
Conv1D_5	[batchsize, 14, 30]	28, 1x1, ReLU, padding
Conv1D_6	[batchsize, 14, 30]	28, 1x1, ReLU, padding
Aggregation_1	[batchsize, 14, 60]	
MaxPooling1D_2	[batchsize, 7, 30]	Pooling size=2
MaxPooling1D_3	[batchsize, 7, 30]	Pooling size=2
Aggregation_2	[batchsize, 7, 60]	
AveragePooling1D	[batchsize, 3, 120]	Pooling size=2
LSTM	[batchsize, 2, 25]	25 neuron, ReLU
Flatten	[batchsize, 360]	
Fully-connected	[batchsize, 2]	4 nodes, softmax

TABLE V
IDEAL FEATURE PARAMETERS OF THE PROPOSED CNN+LSTM MODEL.

Parameters	Layer 1	Layer 2
Total features	28	28
Selected features	14	14
Optimizer	Adam	Adam
learning rate	0.001	0.001
Loss function	Cross-entropy loss	Cross-entropy loss
Epoch	10	10
Batch size	30	30
Activation function	ReLU	ReLU
Cross validation	2 k-fold	2 k-fold
Learnanble parameters	8260	6648

TABLE VI
FEATURE DESCRIPTION OF ICS-SCADA DATASET.

Feature	Description
PA1:VH-PA3:VH	Phase A-C voltage phase angle
PM1:V-PM3:V	Phase A-C voltage magnitude
PA4:IH-PA6:IH	Phase A-C voltage current phase angle
PM4:I-PM6:I	Phase A-C current magnitude
PA7:VH-PA9:VH	Pos.-neg.-zero voltage phase angle
PM7:V-PM12:V	Pos.-neg.-zero voltage magnitude
PA10:VH-PA12:VH	Pos.-neg.-zero current phase angle
PM10:V-PM12:V	Pos.-neg.-zero current magnitude
F	Frequency for relays
DF	Frequency delta (df/dt) for relays
PA:Z	Apparent impedance seen by relays
PA:ZH	Apparent impedance angle seen by relays
S	Status flag for relays

network environment traffic activity, recent and prevalent attacks like eavesdropping, brute force SSH, denial of service (DoS), web attack, user to remote (U2R), man-in-the-middle, brute force FTP, remote to local attack (R2L), Heartbleed, DDos, botnet and Infiltration created for cybersecurity intrusion detection systems [21], [59], [60]. IDS development has been quite challenging in specialized scenarios like the industrial internet of things, particularly SCADA, due to the unavailability of system-specific datasets [48]. Hence, the development of testbeds targeted at this specific industrial control systems (ICS) [48], [52]. These advances aimed at resolving the lack of ICS datasets yielded results by providing datasets like the ICS-SCADA [52] dataset. This dataset was created using SCADA system testbeds and is for SCADA cybersecurity research. See Tables VI for features of the dataset.

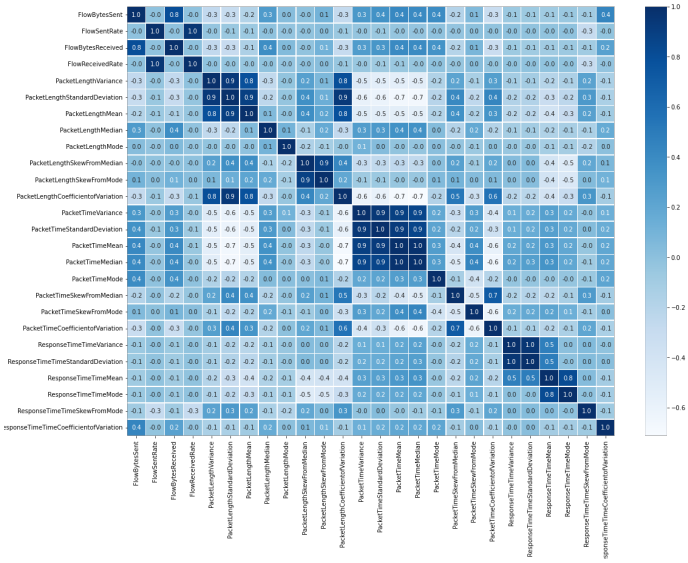

Fig. 5. Correlation matrix showing highly correlated features.

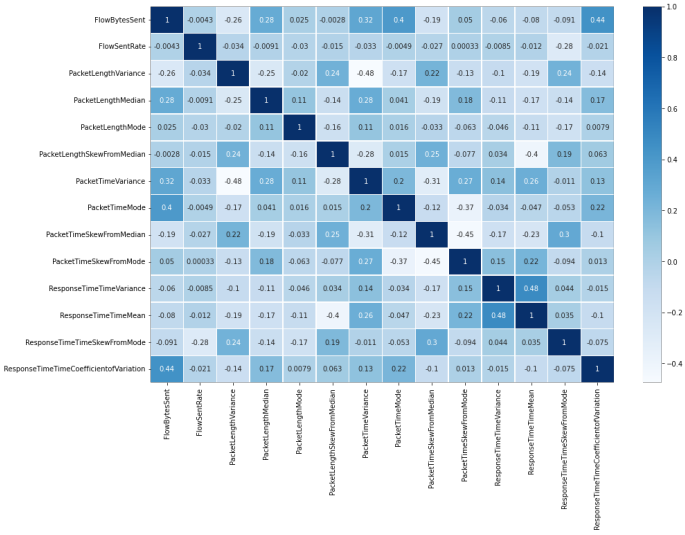

Fig. 6. Result of PCC validating the feature importance selection.

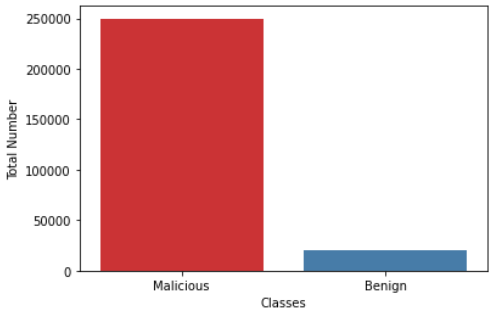

Fig. 7. Dataset distribution showing the malicious class with majority instances and benign class with minority instances.

Data preprocessing is essential for obtaining good quality data before feeding it into the proposed model. Preprocessing, which includes data cleansing and normalization, was done to assure the integrity of the data. Because generating the dataset was by using standard web browsing behaviors for benign-DoH traffic and domain name service channeling procedures for malicious-DoH traffic, [29], it contains high-dimensional features.

1) Data cleansing and normalization: The original dataset consisted of 28 features, some of which are non-contributing. Data cleansing eliminated irrelevant features leaving a balance of 14 out of 28 features. Also cleaned were empty values, NaN values, and infinity (∞). The mean was to fill in blanks and fields with infinity (∞) values in a column. It is to ensure that the model only receives valid data. The Pearson's correlation coefficient (PCC) was applied to the dataset because it comprises strongly associated features, as shown in the correlation matrix in Fig. 5. This approach was required since it reduces over-fitting. Variables with a high correlation value at the threshold of ± 0.7 were selected using the PCC for consecutive variables with a correlation score between -1 and 1, as shown in equation 10. It helps to ensure that only the relevant features are selected, improving the model's performance. The selection of correlated variables with a criterion of ± 0.7 is as in Fig. 6.

$$Q = \frac{\sum (x_i - \hat{x})(y_i - \hat{y})}{\sqrt{\sum (x_i - \hat{x})^2 (y_i - \hat{y})^2}}, \tag{10}$$

where Q represents the PCC, x_i represents the content of the variables in the dataset, $\hat{x}$ represents the mean values of the x variables, y_i represents the sample variables, and $\hat{y}$ represents the mean values of the y variables.

Min-max scaler was the industry standard for measuring all data features between values [0, 1] or [-1, 1]. It is to allow for the scaling of features that are stable in the face of outliers. Following negative values in the data features, rescaling brought it up to unit variance. The SMOTE technique is for data balancing since the data contains an uneven distribution of samples of the classes (total of a group is higher than the other class, (see Fig. 7). The majority class of samples will always skew in an unequal dataset. In an unequal dataset, the majority class of samples will always be favored, resulting in an anomalous dataset and, as a result, low model performance on the minority class; yet, the achievement of the minority class is substantial. This strategy works because it creates new plausible data similar to existing minority class samples. However, due to the potentiality of sample ambiguity as a result of overlapping the classes without consideration for the majority class, this technique should be used with caution.

SMOTE technique is a linear combination of two or more similar samples from minority class (a) and (a^R) defined as:

$$S = a + u.\ (a^R - a), \tag{11}$$

with $0 \leq \theta \leq 1$; a^{R},
randomly chosen among the minority class nearest neighbors of a.

F. Experimental Environment

The suggested system was trained and tested on Google Colab using various Keras and Scikit-learn libraries. We used 499,672 data samples, 70% for the training and the remaining 30% for testing. Furthermore, 20% of the training set for validating the scheme during training (see Fig. 2). Keras's $modelcheckpoint$ callback function was to track and halt model training when the validation accuracy no longer improved, allowing the best model. This method can also be on a minicomputer; a high-level execution computer with GPU may not be necessary. All experimentation can be with NVIDIA GeForce GTX 1050 and 8GB VRAM, with a Windows 10 64-bit operating system.

IV. Performance Evaluation

A. Parameter Metrics

This section demonstrates the achievement of the presented scheme in classifying network traffic (DoH or NonDoH) and anomaly detection (benign or malicious) in the network traffic. In machine learning, direct (machine/environment dependent) and indirect (FLOP: Float-point operations) metrics are used in measuring computational complexity [61]. Using performance assessment measures represented by equations (12), (13), (14), the AUC and (16). The suggested hybrid model for efficient classification and characterization (detection) performance was compared to research by [42], [43], confusion matrix measurement as shown in Table VII, and computation time.

AUC is a metric that summarizes performance across all classification parameters. It ranges in value from 0 to 1. It is to assess the classification accuracy of a model regardless of the categorization criterion employed. A model with 100% incorrect classification will have an AUC value that tends to 0.0, while the one with 100% correct classifications will tend to 1.0.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{12}$$

$$TruePositiveRate = \frac{TP}{TP + FN} = Recall = Sensitivity, \tag{13}$$

$$PositivePredictiveValue\ (Precision) = \frac{TP}{TP + FP}, \tag{14}$$

$$F_1 = \frac{2(Precision.Recall)}{Precision + Recall}. \tag{15}$$

Positioning FN, TP, TN, and FP represent false negative, true positive, true negative, and false positive, respectively.

Another evaluation metrics considered in this study is the MCC. It is to assess the reliability of the accuracy of the classification. It is useful when desired to have a metric that is not affected by unbalanced datasets [31]. The disadvantage of relying on F1-score is that it can result in overoptimistic inflated results, especially on imbalanced datasets. To address this, the authors of [31] provided a complete study and explanation for MCC as a viable alternative. MCC has a value range of -1 to +1, signifying cases of perfect misclassification and perfect classification, respectively. MCC is mathematical as in (16).

$$MCC = \tag{16}$$

$$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}}, \tag{17}$$

TABLE VII
Confusion Matrix Measurement.

Confusion matrix		
	Predicted negative class	Predicted positive class
Actual (-) class	Right benign (TN)	Wrong malicious (FP)
Actual (+) class	Wrong benign (FN)	Right malicious (TP)

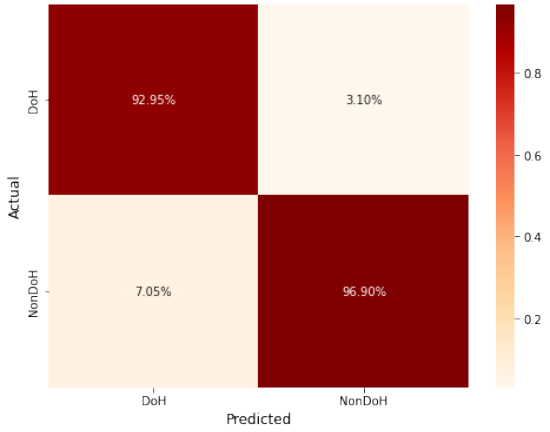

Fig. 8. Confusion matrix of network traffic classification [DoH/NonDoH].

B. Proposed Model Performance Evaluation

The proposed model is excellent for detecting classification and detection. The proposed scheme can classify encoded network traffic as DoH or NonDoH with high precision, recall, and F1-score values. However, there is a need to improve the classification ability of encoded traffic. The traffic classification accuracy needs to be improved compared to the characterization accuracy. Fig. 8 shows the confusion matrix of the proposed CNN-LSTM scheme for encoded traffic classification, providing an error matrix between the predicted and the actual. It is evident from the result that the proposed CNN-LSTM scheme is promising in detecting and classifying the encoded traffic with minimal misclassification. It is important to note that a more sophisticated high-volume application-layer attack characterizes the encoded DNS traffic.

Subsequently, Fig. 9 is the confusion matrix of the proposed models' performance in characterizing benign and malicious network traffic with 98.88% and 99.68%, respectively.

However, to resolve the possibility of ambiguity of samples and bias in misleading results due to data balancing by the SMOTE technique, the proposed model evaluated the dataset without balancing. See section Fig. 10, showing high precision of 95%.67 for Benign and 99.45% for malicious detection. The

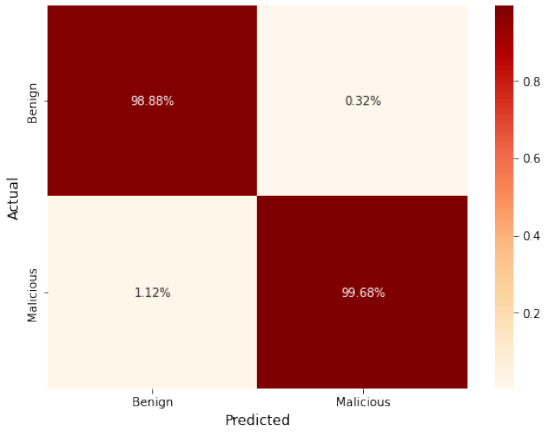

Fig. 9. Confusion matrix of anomaly detection [benign/malicious].

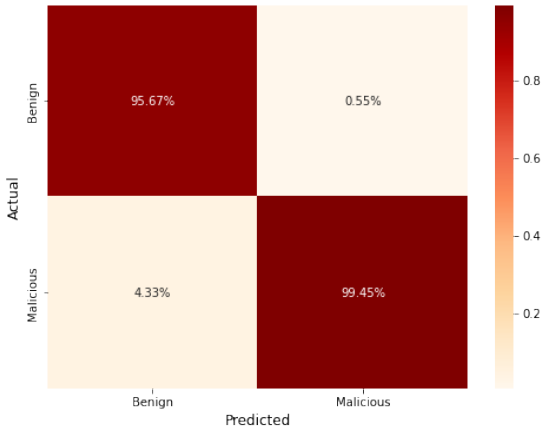

Fig. 10. Confusion matrix of anomaly detection [benign/malicious] without data balancing.

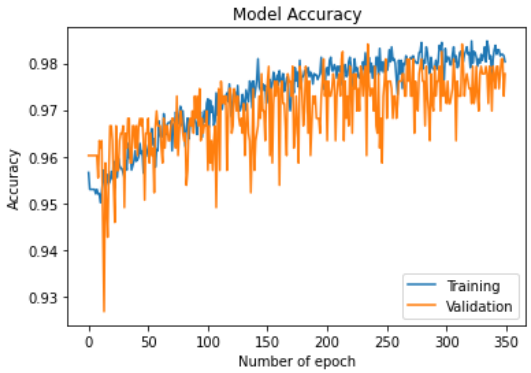

Fig. 11. Accuracy graph showing the proposed models' performance on ICS-SCADA dataset.

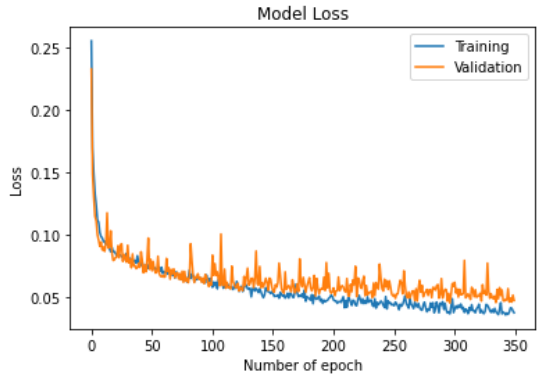

Fig. 12. Loss graph showing the proposed models' performance on ICS-SCADA dataset.

model performance on both balanced and unbalanced datasets illustrates the model's performance regarding

Integrating CNN and LSTM networks significantly improves the model's ability to recognize intricate features in both spatial and sequential domains. The hybrid architecture excels in capturing complex patterns, leading to high-accuracy performance and a deep understanding of its behavior and misclassification instances. The study confirms the effectiveness of the hybrid CNN-LSTM classification model in handling complex data patterns, making it a promising approach for various classification tasks in diverse domains. The insights gained from this study pave the way for future enhancements and applications of hybrid models in real-world scenarios.

C. Reliability Test for the Proposed Models' Performance

The proposed hybrid DL model evaluated the ICS-SCADA, NSL-KDD, and CICIDS2017 datasets to validate the reliability of the suggested model. The proposed model had significant performance in improved detection accuracy and reduction in computation time. Tables VIII and IX show the significance of the model performance for train and test simulations. Figs. 11 and 12 represent the accuracy and loss performance of the proposed model in the ICS-SCADA dataset. It demonstrates the model training performance over epochs; this is important for illustrating the achievement of high accuracy. Fig. 13 is the confusion matrix showing the performance of the proposed model on the ICS-SCADA binary and three class datasets. It shows that the proposed model is significant in the detection of attacks, benign and zero-day attacks. We demonstrate its applicability for intrusion detection in IIoT (SCADA) communication networks. Also, Fig. 14 demonstrates the outperformance of the proposed model over the existing studies of [42] and [43] in accuracy and execution time on similar datasets. Further validating the suitability of the proposed model on multi-class datasets.

D. Comparative Analysis of the Proposed Model Across State-of-the-art Datasets

Table X shows the proposed model's performance compared to state-of-the-art models in recent studies. It is the classification and detection reports based on training and testing time, accuracy, precision, recall, AUC parameter metrics, and datasets used regarding the relevance of current cyber-security and SCADA datasets. In comparison with the study by [42] on the CICIDS2017 dataset with a detection accuracy of 99.03%, training time (85255.63 s) and testing time (15313.1036 s). The proposed model outperformed utilizing the same dataset

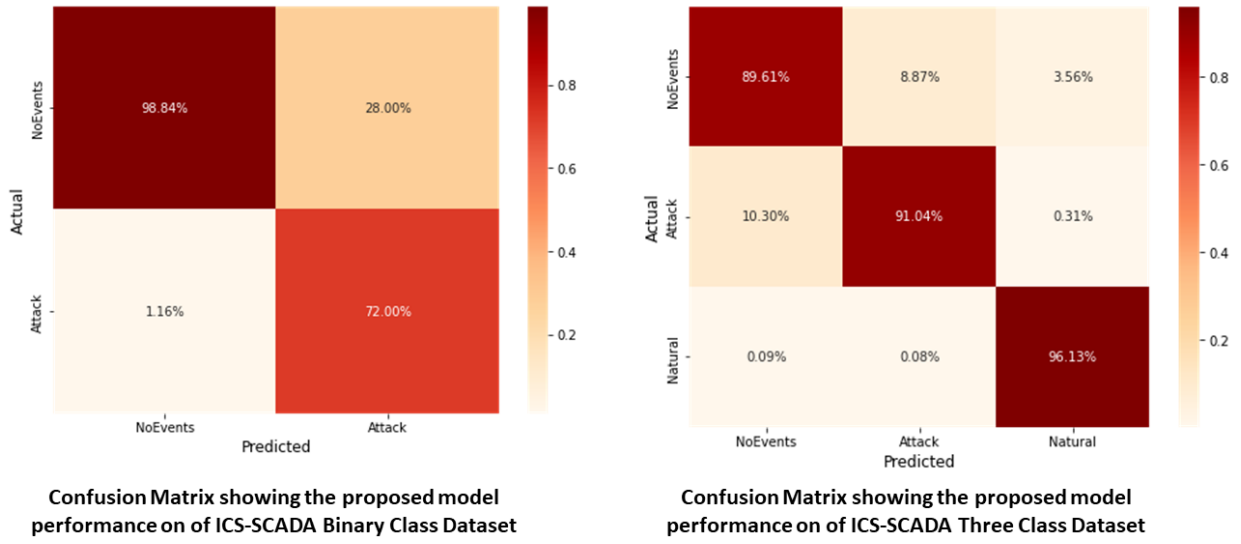

Fig. 13. Confusion matrix showing the proposed model performance on ICS-SCADA binary and three class datasets.

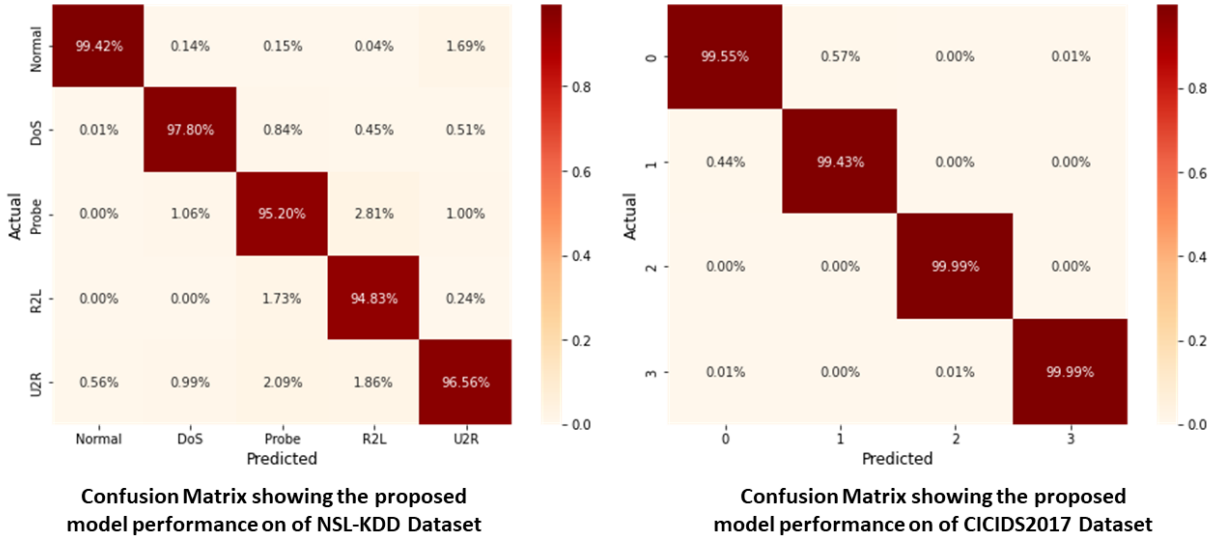

Fig. 14. Confusion matrix showing the proposed model performance on NSL-KDD and CICIDS2017 datasets.

TABLE VIII
TRAINING MODEL PERFORMANCE ACROSS EVALUATED STATE-OF-THE-ART DATASETS.

Dataset	Training time (s)	Train accuracy (%)	Train precision (%)	Train recall (%)	Train F1 score	Train AUC
NSL-KDD [59]	500.51	97.51	97.59	97.44	97.59	0.9969
CICIDS2017 [60]	589.40	99.60	99.62	99.58	99.40	0.9999
CIRA-CIC-DoHBrw-2020 [21]	745	98.32	98.32	98.32	98.31	0.9975
ICS-SCADA [52]	265.17	98.05	98.30	98.30	98.23	0.9972

TABLE IX
TESTING MODEL PERFORMANCE ACROSS EVALUATED STATE-OF-THE-ART DATASETS.

Dataset	Testing time (s)	Test accuracy (%)	Test precision (%)	Test Recall (%)	Test F1 score	Test AUC
NSL-KDD [59]	0.000136	97.79	97.85	97.73	97.92	0.9969
CICIDS2017 [60]	0.000297	99.74	99.74	99.74	99.74	0.9999
CIRA-CIC-DoHBrw-2020 [21]	0.000324	99.28	99.28	99.75	99.23	0.9975
ICS-SCADA [52]	0.000252	99.05	99.30	99.30	99.19	0.9972

TABLE X
MODEL COMPARISON OF EXISTING APPROACHES UTILIZING A HYBRID OF THE CONVOLUTIONAL NEURAL NETWORK AND LONG SHORT-TERM MEMORY (CNN+LSTM).

Model/ Author	Training time (s)	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	Dataset
NSGA-II-aJG and CNN+LSTM [42]	15313.103	99.03	99.26	99.35	99.36	CICIDS2017
CNN-LSTM [43]	542	89.23	86.86	88.58	88.58	NSL-KDD
CNN-LSTM [44]	-	99.78	-	-	-	KDD99
DL-IDS (CNN+LSTM) [45]	-	98.67	97.21	93.32	93.32	CICIDS2017
This study (CNN+LSTM)	265.17	99.35	99.30	99.28	99.30	ICS-SCADA
This study (CNN+LSTM)	589.40	99.74	99.74	99.74	99.74	CICIDS2017
This study (CNN+LSTM)	500.51	97.51	97.51	97.51	97.51	NSL-KDD

with an improved detection accuracy of 99.73%, reduced training time (589.40), and testing time (0.000297).

The proposed model showed significant improvement in evaluating the NSL-KDD dataset for multiple attack types, recording a detection accuracy of 97.51% with minimal training and testing time of 500.51 s and 0.000136 s, respectively. Also, the proposed model outperformed all other models in a combined advantage of parameter metrics, as shown in the table. However, the study by [43] using the NSL-KDD dataset recorded a computation time of 61s with poor accuracy (89.23%) and precision (86.86%), recall (88.58%) with no consideration for AUC value. To reduce computation time, the authors transformed the standardized dataset into an image to achieve a swifter model. This approach does not apply to real scenarios, hence, considered not a fair comparison.

E. Resilience Performance of the Proposed Model: Computation Time, Accuracy, and MCC

Following the comparative analysis, an in-depth investigation of the performance of the proposed hybrid DL model shows improved and better performance in a combined advantage of parameter metrics over compared recent studies. On the recent high dimensional CIRA-CIC-DoHBrw-2020 cyber-security dataset, the proposed scheme demonstrated efficiency in classifying and characterizing attacks in an encoded SCADA network traffic communication with high detection accuracy, precision, recall, and AUC. As explained in Section IV-A, the resilience and reliability of the proposed model was investigated using the MCC to resolve the drawback due to reliance on the F1-score and checkmate the tendency for overfitting. Table XI shows that the MCC performance of the proposed model was consistent on all datasets options. This metric authenticates the detection accuracy of the proposed model, confirming its non-bias due to overfitting. It also validates the applicability of the proposed model in real scenarios with unbalanced data. The proposed model also shows capability in detecting multiple attack types with precision, as witnessed in the evaluation results of NSL-KDD and CICIDS2017. Also, in a SCADA environment, the proposed model outperformed with a combined advantage of high precision of 99.30%, accuracy (99.35%), recall (99.30%), AUC (0.9972), and the minimal training and testing time of 265.17 s and 0.000252 s respectively, on the high-dimensional ICS-SCADA dataset.

TABLE XI
MODEL PERFORMANCE VALIDATION USING THE MATHEWS CORRELATION COEFFICIENT (MCC) ACROSS EVALUATED DATASETS.

Dataset	Test MCC
NSL-KDD [59]	0.9724
CICIDS2017 [60]	0.9965
CIRA-CIC-DoHBrw-2020 [21]	0.9975
ICS-SCADA [52]	0.8864

V. CONCLUSION

This paper proposes a CNN-LSTM hybrid model to accurately categorize network traffic as DoH or NonDoH, indicating whether the traffic is benign, malicious, or zero-day. CNN was to extract key characteristics from network traffic; for effective classification, these traits constitute the input to the LSTM. Max and average pooling enabled the extraction of significant features from the feature map according to filter size and strides. It aided the reduction in computation costs. The model's batch normalization enables faster convergence with minimal computing complexity, regardless of the amount of network data. These features enabled the swiftness of the proposed model, thereby improving its stability and minimizing execution time. Dropout layers helped protect the model from overfitting. As a result, it enhanced the network's robustness. The findings also showed that the proposed hybrid model decreases computing complexity while increasing precision and accuracy. The simulation result demonstrates the suggested approach's efficiency, which shows significant detection accuracy, precision, recall rate, the least execution time, AUC, and MCC across evaluated datasets. The suggested hybrid model outperformed on large and sparse datasets compared to previous models. As a result of the reliability test using four (4) publicly available datasets, the suggested model applies to any network system in real-time. A future research direction is introducing Gaussian noise to the model and investigating the computational cost.

REFERENCES

[1] K. O. Akpinar and I. Ozcelik, "Analysis of machine learning methods in EtherCAT-based anomaly detection," *IEEE Access*, vol. 7, pp. 184365–184374, 2019.

[2] H. H. Pajouh, R. Javidan, R. Khayami, A. Dehghantanha, and K.-K. R. Choo, "A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks," *IEEE Trans. Emerg. Topics Comput.*, vol. 7, no. 2, pp. 314–323, 2019.

[3] A. Azmoodeh, A. Dehghantanha, and K.-K. R. Choo, "Robust malware detection for Internet of (battlefield) things devices using deep eigenspace learning," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 1, pp. 88–95, 2019.

[4] T. Saba, T. Sadad, A. Rehman, Z. Mehmood, and Q. Javaid, "Intrusion detection system through advance machine learning for the Internet of things networks," *IT Professional*, vol. 23, no. 2, pp. 58–64, 2021.

[5] M. Kordestani and M. Saif, "Observer-based attack detection and mitigation for cyberphysical systems: A review," *IEEE Systems. Man. and Cybern. Mag.*, vol. 7, no. 2, pp. 35–60, 2021.

[6] U. S. Musa, S. Chakraborty, M. M. Abdullahi, and T. Maini, "A review on intrusion detection system using machine learning techniques," in *Proc. IEEE ICCCIS*, 2021.

[7] G. Amaizu, C. Nwakanma, S. Bhardwaj, J. Lee, and D. Kim, "Composite and efficient DDoS attack detection framework for B5G networks," *Comput. Netw.*, vol. 188, p. 107871, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128621000438

[8] H. Tran-Dang, S. Bhardwaj, T. Rahim, A. Musaddiq, and D.-S. Kim, "Reinforcement learning based resource management for fog computing environment: Literature review, challenges, and open issues," *J. Commun. Netw.*, vol. 24, no. 1, pp. 83–98, 2022.

[9] S. Hosseini and M. Azizi, "The hybrid technique for DDoS detection with supervised learning algorithms," *Comput. Netw.*, vol. 158, pp. 35–45, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128618306881

[10] D.-S. Kim and H. Tran-Dang, "Industrial sensors and controls in communication networks," in *Comput. Commun. Netw.* Springer, 2019.

[11] S. N. Mohan, G. Ravikumar, and M. Govindarasu, "Distributed intrusion detection system using semantic-based rules for SCADA in smart grid," in *Proc. IEEE TD*, 2020.

[12] A. A. el Kalam, "Securing SCADA and critical industrial systems: From needs to security mechanisms," *International J. Critical Infrastructure Protection*, vol. 32, p. 100394, 2021.

[13] Z. Yan, H. Li, S. Zeadally, Y. Zeng, and G. Geng, "Is DNS ready for ubiquitous Internet of things?" *IEEE Access*, vol. 7, pp. 28 835–28 846, 2019.

[14] J. Yang *et al.*, "Ultra-reliable communications for industrial Internet of things: Design considerations and channel modeling," *IEEE Netw.*, vol. 33, no. 4, pp. 104–111, 2019.

[15] T. Taleb, I. Afolabi, and M. Bagaa, "Orchestrating 5G network slices to support industrial internet and to shape next-generation smart factories," *IEEE Netw.*, vol. 33, no. 4, pp. 146–154, 2019.

[16] P. Kehl *et al.*, "Comparison of 5G enabled control loops for production," in *Proc. IEEE PIMRC*, 2020.

[17] Z. Lü, Y. Lü, M. Yuan, and Z. Wang, "A heterogeneous large-scale parallel SCADA/DCS architecture in 5G OGCE," in *Proc. IEEE CISP-BMEI*, 2017.

[18] A. Rostami, "Private 5G networks for vertical industries: Deployment and operation models," in *Proc. IEEE 5GWF*, 2019.

[19] E. C. Strinati *et al.*, "Beyond 5G private networks: The 5G CONNI perspective," in *Proc. IEEE GCWkshps*, 2020.

[20] Y.-i. Choi and J. H. Kim, "Reliable data transmission in 5G network using access traffic steering method," in *Proc. IEEE ICTC*, 2020.

[21] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. H. Lashkari, "Detection of DoH tunnels using time-series classification of encrypted traffic," in *Proc. IEEE DASC*, 2020.

[22] A. M. Hassan and A. I. Awad, "Urban transition in the era of the Internet of things: Social implications and privacy challenges," *IEEE Access*, vol. 6, pp. 36 428–36 440, 2018.

[23] G. Lencse, "Benchmarking authoritative DNS servers," *IEEE Access*, vol. 8, pp. 130 224–130 238, 2020.

[24] G. Zhao, K. Xu, L. Xu, and B. Wu, "Detecting APT malware infections based on malicious DNS and traffic analysis," *IEEE Access*, vol. 3, pp. 1132–1142, 2015.

[25] A. Satoh, Y. Nakamura, Y. Fukuda, K. Sasai, and G. Kitagata, "A cause-based classification approach for malicious DNS queries detected through blacklists," *IEEE Access*, vol. 7, pp. 142 991–143 001, 2019.

[26] Z. Liang, T. Zang, and Y. Zeng, "MalPortrait: Sketch malicious domain portraits based on passive DNS data," in *Proc. IEEE WCNC*, 2020.

[27] T. Mahjabin, Y. Xiao, T. Li, and C. L. P. Chen, "Load distributed and benign-bot mitigation methods for IoT DNS flood attacks," *IEEE Internet Things J.*, vol. 7, no. 2, pp. 986–1000, 2020.

[28] R. Chandramouli and S. Rose, "Secure domain name system (DNS) deployment guide," *NIST Special Publication*, vol. 800, pp. 81–2, 2006.

[29] M. MontazeriShatoori, L. Davidson, G. Kaur, and A. H. Lashkari, "Detection of DoH tunnels using time-series classification of encrypted traffic," in *Proc. IEEE CyberSciTech*, 2020.

[30] Y. Wei *et al.*, "AE-MLP: A Hybrid deep learning approach for DDoS detection and classification," *IEEE Access*, vol. 9, pp. 146 810–146 821, 2021.

[31] D. Chicco and G. Jurman, "The advantage of the Mathews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 6, pp. 1–13, 2020.

[32] D. Bhamare *et al.*, "Cybersecurity for industrial control systems: A survey," *Compu. Security*, vol. 89, p. 101677, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404819302172

[33] R. R. Barbosa, R. Sadre, and A. Pras, "Difficulties in modeling SCADA traffic: A comparative analysis," in *Proc. PAM*, 2012.

[34] C.-Y. Lin and S. Nadjm-Tehrani, "Understanding IEC-60870-5-104 traffic patterns in SCADA networks," in *Proc. ACM CPSS*, 2018.

[35] C. Yue, L. Wang, D. Wang, R. Duo, and X. Nie, "An ensemble intrusion detection method for train ethernet consist network based on CNN and RNN," *IEEE Access*, vol. 9, pp. 59 527–59 539, 2021.

[36] N. Moustafa, B. Turnbull, and K.-K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4815–4830, 2019.

[37] A. Abbasi *et al.*, "ElStream: An ensemble learning approach for concept drift detection in dynamic social big data stream learning," *IEEE Access*, vol. 9, pp. 66 408–66 419, 2021.

[38] S. K. Singh and P. K. Roy, "Detecting malicious DNS over HTTPS traffic using machine learning," in *Proc. IEEE 3ICT*, 2020.

[39] R. Alenezi and S. A. Ludwig, "Classifying DNS tunneling tools for malicious doh traffic," in *Proc. IEEE SSCI*, 2021.

[40] F. Binhao, H. Hong, and Z. Ziyun, "Improve the application of xgbdt in network abnormal traffic detection," in *Proc. IEEE ICESIT*, 2021.

[41] L. F. G. Casanova and P.-C. Lin, "Generalized classification of DNS over HTTPS traffic with deep learning," in *Proc. IEEE APSIPA ASC*, 2021.

[42] M. Roopak, G. Y. Tian, and J. Chambers, "An intrusion detection system against DDoS attacks in IoT networks," in *Proc. IEEE CCWC*, 2020.

[43] L. Karanam, K. K. Pattanaik, and R. Aldmour, "Intrusion detection mechanism for large scale networks using CNN-LSTM," in *Proc. IEEE DeSE*, 2020.

[44] K. Praanna, S. Sruthi, K. Kalyani, and A. S. Tejaswi, "A CNN-LSTM model for intrusion detection system from high dimensional data," *J. Inf. Comput. Sci*, vol. 10, pp. 1362–1370, 2020.

[45] P. Sun *et al.*, "DL-IDS: Extracting features using CNN-LSTM hybrid network for intrusion detection system," *Security Commun.n Netw.*, vol. 2020, 2020.

[46] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 41–50, 2018.

[47] M. Abdel-Basset, H. Hawash, R. K. Chakrabortty, and M. J. Ryan, "Semi-supervised spatiotemporal deep learning for intrusions detection in IoT networks," *IEEE Internet Things J.*, vol. 8, no. 15, pp. 12251–12265, 2021.

[48] M. Teixeira, T. Salman, M. Zolanvari, R. Jain, N. Meskin, and M. Samaka, "SCADA system testbed for cybersecurity research using machine learning approach," *Future Internet*, vol. 10, no. 8, p. 76, Aug 2018. [Online]. Available: http://dx.doi.org/10.3390/fi10080076

[49] M. A. Teixeira, M. Zolanvari, K. M. Khan, R. Jain, and N. Meskin, "Flow-based intrusion detection algorithm for supervisory control and data acquisition systems: A real-time approach," *IET Cyber-Physical Systems: Theory & Applications*, vol. 6, no. 3, pp. 178–191, 2021.

[50] D. Upadhyay, J. Manero, M. Zaman, and S. Sampalli, "Intrusion detection in SCADA based power grids: Recursive feature elimination model with majority vote ensemble algorithm," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2259–2574, 2021.

[51] Y. Ouyang, B. Li, Q. Kong, H. Song, and T. Li, "FS-IDS: A novel few-shot learning based intrusion detection system for SCADA networks," in *Proc. IEEE ICC*, 2021.

[52] U. Adhikari, S. Pan, T. Morris, R. Borges, and J. Beave, "Industrial control system (ICS) cyber attack datasets," *Datasets used in the Experimentation.*, 2019. [Online]. Available: https://sites. google. com/a/uah. edu/tommy-morris-uah/ics-data-sets

[53] K. Sayed and H. A. Gabbar, "SCADA and smart energy grid control automation," in *Smart Energy Grid Engineering*. Elsevier, 2017.

[54] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, "A survey of network-based intrusion detection datasets," *Compu. Security*, vol. 86, pp. 147–167, 2019.

[55] M. Hajiaghayi and E. Vahedi, "Code failure prediction and pattern extraction using LSTM networks," in *Proc. IEEE BigDataService*, 2019.

[56] K. Yan *et al.*, "Multi-step short-term power consumption forecasting with a hybrid deep learning strategy," *Energies*, vol. 11, no. 11, 2018. [Online]. Available: https://www.mdpi.com/1996-1073/11/11/3089
[57] L. A. C. Ahakonye, C. I. Nwakanma, J.-M. Lee, and D.-S. Kim, "Efficient classification of enciphered SCADA network traffic in smart factory using decision tree algorithm," *IEEE Access*, vol. 9, pp. 154892–154901, 2021.
[58] "CIRA-CIC-DoHBrw-2020 dataset," 2020. [Online]. Available: https://www.unb.ca/cic/datasets/dohbrw-2020.html
[59] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE CISDA*, 2009.
[60] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," *ICISSp*, vol. 1, pp. 108–116, 2018.
[61] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in *Proc. ECCV*, 2018.

Love Allen Chijioke Ahahkonye (Student Member, IEEE) received the B.Sc. degree in mathematics/computer science from the University of Port Harcourt, Port Harcourt, Nigeria, in 2001, and the M.Sc. degree in Information Technology from the Federal University of Technology, Owerri, Nigeria, in 2016. She is currently working toward the Ph.D. degree in IT-convergence engineering from the Kumoh National Institute of Technology, Gumi, South Korea. She is currently a Full-time Researcher with Networked Systems Lab, IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea. She has a background in the Nigerian oil and gas sector, where she was a Network and System Administrator, from 2002 to 2016, and a Logistics Superintendent with the Nigerian Petroleum Development Company, from 2017 to 2019. Her research interests focus on the application of artificial intelligence (AI) to Industrial Internet of Things (IIoT) SCADA systems, particularly in the areas of intrusion/anomaly detection, cybersecurity, fault detection, and manufacturing execution systems.

Gabriel Chukwunonso Amaizu received his bachelor's degree in computer application from the prestigious Bangalore University, India in 2017. He received his master's degree in the Department of IT Convergence Engineering, Kumoh National Institute of Technology, South Korea in 2021. From 2021 to 2022, he worked as an AI+Security researcher with the ICT Convergence Research Center, Kumoh National Institute of Technology, South Korea. Currently, he is a Ph.D. student in IT Convergence Engineering from the Kumoh National Institute of Technology, Gumi, South Korea. His research interests include network security, machine learning, systems design, and industrial IoT.

Cosmas Ifeanyi Nwakanma (M'19) received the Ph.D. degree in IT Convergence Engineering from Kumoh National Institute of Technology, Gumi, South Korea, in 2022. He is a Senior Research Fellow with the ICT-Convergence Research Center, Kumoh National Institute of Technology. From 2006 to 2009, he was with First Bank of Nigeria PLC, and from 2009 to 2019 was a Lecturer with the Department of Information Technology, the Federal University of Technology Owerri, Nigeria. From 2019 to 2022, he was a Senior Graduate Research Assistant with the Networked Systems Lab, Kumoh National Institute of Technology. Since 2022, he has been a Senior Researcher with the ICT-CRC. His research interests include explainable artificial intelligence (XAI) applications in the Internet of Things (IoT) for various domains, including smart factories, farms, homes, vehicles, and the Metaverse. Dr. Nwakanma is a Member of the Computer Professionals Registration Council of Nigeria (CPN), Nigeria Society of Engineers (NSE), and registered by the Council for the Regulation of Engineering in Nigeria (COREN)

Jae-Min Lee (Member, IEEE) received the Ph.D. degree in Electrical and Computer Engineering from Seoul National University, Seoul, Korea, in 2005. He is an Associate Professor with the Department of IT Convergence Engineering, Kumoh National Institute of Technology, Gumi, South Korea. From 2005 to 2014, he was a Senior Engineer with Samsung Electronics, Suwon, Korea. From 2015 to 2016, he was a Principal Engineer with Samsung Electronics, Suwon, Korea. Since 2017, he has been an Associate Professor with the School of Electronic Engineering and Department of IT-Convergence Engineering, Kumoh National Institute of Technology, Gyeongbuk, Korea. His current main research interests include smart IoT convergence applications, industrial wireless control networks, UAVs, metaverse, and blockchain.

Dong-Seong Kim (SM'14) received the Ph.D. degree in electrical and computer Engineering from Seoul National University, Seoul, South Korea, in 2003. From 1994 to 2003, he was a full-time Researcher with ERC-ACI, Seoul National University. From 2003 to 2005, he was a Postdoctoral Researcher with the Wireless Network Laboratory, School of Electrical and Computer Engineering, Cornell University, New York, NY, USA. From 2007 to 2009, he was a Visiting Professor with the Department of Computer Science, University of California,-Davis, CA, USA. He is currently the Dean of the Industrial Academic Cooperation Foundation and the Director of the ICT Convergence Research Center (ITRC and NRF Advanced Research Center Program) supported by the Korean Government at Kumoh National Institute of Technology, Gumi, South Korea. His current research interests are real-time IoT and smart platforms, industrial wireless control networks, networked embedded systems, field buses, metaverse, and blockchain. Dr. Kim is a Senior Member of ACM.