IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 23 December 2023, accepted 3 January 2024, date of publication 8 January 2024,
date of current version 16 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3350978

RESEARCH ARTICLE

Enhanced CNN-LSTM Deep Learning for SCADA IDS Featuring Hurst Parameter Self-Similarity

ASAAD BALLA[1], MOHAMED HADI HABAEBI[1], (Senior Member, IEEE), ELFATIH A. A. ELSHEIKH[2], (Member, IEEE), MD. RAFIQUL ISLAM[1], (Senior Member, IEEE), FAKHER ELDIN MOHAMED SULIMAN[2], (Member, IEEE), AND SINIL MUBARAK[1]

[1]Department of Electrical and Computer Engineering, International Islamic University Malaysia, Kuala Lumpur 53100, Malaysia
[2]Department of Electrical Engineering, College of Engineering, King Khalid University, Abha 61421, Saudi Arabia

Corresponding author: Mohamed Hadi Habaebi (habaebi@iium.edu.my)

The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work through the Research Group Program under Grant Number (R.G.P.1/221/44).

ABSTRACT Supervisory Control and Data Acquisition (SCADA) systems are crucial for modern industrial processes and securing them against increasing cyber threats is a significant challenge. This study presents an advanced method for bolstering SCADA security by employing a modified hybrid deep learning model. A key innovation in this work is integrating the Self-similarity Hurst parameter into the dataset alongside a CNN-LSTM model, significantly boosting the Intrusion Detection System's (IDS) capabilities. The Hurst parameter, which quantifies the self-similarity in a dataset, is instrumental in detecting anomalies. Our in-depth analysis of the CICIDS2017 dataset sheds light on contemporary attack patterns and network traffic behaviors. The CNN-LSTM architecture was substantially altered by adding multiple convolutional layers with progressively increasing filters, batch normalization for stable training, and dropout layers for regularization. Principal Component Analysis (PCA) was applied for dimensionality reduction, thereby optimizing the dataset. Test results demonstrate the superior performance of the model incorporating the Hurst parameter, achieving 95.21% accuracy and 82.59% recall, significantly surpassing the standard model. The inclusion of the Hurst parameter marks a substantial advancement in identifying emerging threats, while architectural improvements to the CNN-LSTM model led to more robust and accurate intrusion detection in industrial control settings.

INDEX TERMS Deep learning, intrusion detection system, supervisory control and data acquisition, self-similarity, Hurst parameter, binary classification.

I. INTRODUCTION

Supervisory Control and Data Acquisition (SCADA) systems play a crucial role in modern industrial control systems, serving as a vital component for monitoring and managing various industrial processes. They form the backbone of critical infrastructure sectors like power generation, water treatment, and oil and gas production facilities.

The current state of SCADA systems includes advancements in connectivity, data analytics, and remote access capabilities, allowing for more efficient and centralized management of distributed systems. However, despite these advancements, ensuring the security of SCADA systems poses a significant challenge [1]. The growing prevalence of cyber threats has heightened concerns about the vulnerability of SCADA systems to attacks, leading to a notable increase in research efforts focused on enhancing their security [2].

Along with these advancements, the integration of the Internet of Things (IoT) into SCADA systems has brought additional complexities. These systems, traditionally isolated from the internet and with real-time requirements, now face challenges that impede the implementation of standard security measures. Emerging security issues include complexity, absence of security standards, device heterogeneity, and security vulnerabilities in IoT devices, making safeguarding SCADA systems against cyber threats more challenging [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Leandros Maglaras.

Neural networks (NN), the core building blocks of deep learning (DL), are algorithms modeled after the human brain's structure and function, designed to recognize patterns, and solve complex problems. They consist of interconnected nodes (neurons) organized in layers, with each layer capable of performing specific computations. DL involves using neural networks with many layers (deep architectures) to process and learn from large amounts of data. This advanced form of machine learning enables the identification of intricate patterns and relationships in data, making it highly effective for tasks such as anomaly detection in SCADA systems.

To effectively address these emerging challenges, a comprehensive approach is necessary. This approach must encompass IoT device security, establishing security standards, and integrating multiple devices and protocols into a unified and secure system. Such a strategy is crucial to maintain the confidentiality, integrity, and availability of data and control processes within SCADA systems.

In response to these challenges, a fresh approach is proposed to increase the security of SCADA systems, especially those incorporating IoT. This approach suggests utilizing the Self-similarity Hurst parameter alongside the CNN-LSTM (Convolutional Neural Network, Long Short-Term Memory) to deliver a comprehensive security solution.

The choice of the Self-similarity Hurst parameter is grounded in its proven capability to detect evolving anomalies in time-series data, a common characteristic of cyber threats in SCADA systems. Similarly, the CNN-LSTM model is selected for its ability to capture both spatial and temporal dependencies in data, offering a robust framework for real-time anomaly detection.

In IDSs for SCADA systems, binary classification is employed to categorize network events as either normal or anomalous [4]. This approach is pivotal in identifying potential intrusions, with algorithms analyzing diverse data features to make decisive, binary judgments about each network event's nature.

Long-Short Term Memory (LSTMs) are particularly beneficial in this context due to their proficiency in processing and memorizing long sequences of data [5], a characteristic typical of SCADA systems' operations. By capturing long-term dependencies, LSTMs can detect intricate anomaly patterns that might be missed by other methods, thereby enhancing the efficacy of intrusion detection systems.

To build a data-driven foundation for our security models, we undertook an extensive Exploratory Data Analysis (EDA) on the CICIDS2017 dataset [6]. This rigorous analysis was critical for understanding the dataset's intricacies and nuances, paving the way for the development and implementation of effective security solutions.

The central research question is: "How effective is the combined use of the Self-similarity Hurst parameter and the CNN-LSTM model in detecting anomalies in SCADA systems?" The study contributes by developing a hybrid intrusion detection system that harnesses the strengths of both the Self-similarity Hurst parameter and the CNN-LSTM, aiming to enhance anomaly detection in SCADA systems.

The comprehensive approach we employed in this study successfully achieved the objective of detecting anomalies in SCADA systems. We proposed a hybrid Intrusion Detection System (IDS) that combines the self-similarity approach using the Hurst parameter with a modified CNN-LSTM model, enhancing the system's ability to detect anomalies.

Further enhancement of the IDS was achieved with the PCA method, contributing to a more effective analysis. The model, structured with input, hidden, and output layers, integrated relevant activation functions for optimal performance. Specifically, modifications in the Recurrent Neural Networks Long-Short Term Memory (RNN-LSTM) included adjustments to the LSTM, dense, dropout, and output layers, culminating in a robust solution to address the vulnerabilities of SCADA systems.

This research introduces an innovative approach to enhancing SCADA system security by augmenting the Hurst parameter into the dataset as an additional feature. This novel application, integrated with a CNN-LSTM model, advances anomaly detection capabilities beyond the current scope in cybersecurity literature. While the incorporation of the Hurst parameter itself is not unprecedented, its specific use as an augmenting feature within a deep learning framework represents a significant methodological advancement. This study thereby contributes to the evolving field of AI and cybersecurity, demonstrating a nuanced application of statistical measures in conjunction with deep learning techniques. The implications of this approach are substantial, suggesting a new direction for integrating multidisciplinary methods to enhance the adaptability and efficacy of cybersecurity systems.

The contributions of this research are as follows:

1- We introduce the Self-similarity Hurst parameter into the CICIDS2017 dataset for anomaly detection in SCADA systems using a CNN-LSTM model. This novel approach leverages the parameter's ability to detect evolving anomalies in time-series data, offering a new dimension to intrusion detection methodologies.
2- By refining the CNN-LSTM architecture specifically for SCADA systems, this research enhances the model's capability to capture both spatial and temporal data dependencies, crucial for real-time anomaly detection.
3- The undertaking of an extensive Exploratory Data Analysis (EDA) on the CICIDS2017 dataset sets a solid foundation for this research, providing critical insights into network behaviors and attack patterns.
4- The application of Principal Component Analysis (PCA) for dimensionality reduction optimizes the model's performance, balancing computational efficiency with the complexity of the dataset.
5- The study extends beyond theoretical innovation, demonstrating practical applications in enhancing SCADA system security against evolving cyber threats.

It offers a model that is not only academically robust but also practically applicable in various SCADA environments.

The paper is organized as follows: Section II reviews relevant literature on SCADA system security and the application of deep learning in intrusion detection. Section III details the chosen methodology, including data collection and analysis of the CICIDS2017 dataset. The results and their analysis are discussed in Section IV, and Section V concludes with a summary of the outcomes and future research directions.

II. RELATED WORK

Cyberattack detection in SCADA IoT networks is crucial for ensuring the reliability and security of critical infrastructure systems. One promising approach for detecting intrusions in these networks is the use of self-similarity analysis in combination with machine learning and deep learning techniques. Self-similarity analysis can effectively identify abnormal behavior patterns in the network and, when combined with ML/DL algorithms, can provide a highly effective solution for detecting cyberattacks in SCADA IoT environments. This section provides an overview of recent studies that utilize self-similarity analysis and ML/DL algorithms for intrusion detection in the SCADA IoT, along with an analysis of their limitations and challenges.

Security specialists have formulated various deep learning-based Intrusion Detection Systems (IDSs). For instance, one study [7] explores the combination of signature-based IDS with Long-Short Term Memory (LSTM) to identify Distributed Denial of Service (DDoS) attacks on IoT networks. In contrast, another research [8] investigates the use of the Spider Monkey Optimization (SMO) algorithm and the Stacked-Deep Polynomial Network (SDPN) in enhancing sensor data detection in IoT frameworks, although the complexity of SMO could lead to computational inefficiencies [9].

Additionally, the adoption of Convolutional Neural Networks (CNNs) in IDS research is evident. A particular study [8] applied CNNs to detect specific web protocol intrusions, such as those linked to the Hypertext Transfer Protocol (HTTP). While effective, CNNs require substantial amounts of labeled data for training, which may not always be available. With training on the CICIDS2017 dataset, they underscored the capabilities of deep learning to decipher intricate data and yield reliable results.

An innovative IDS model for Industrial IoT (IIoT) networks that combines CNN with LSTM is described in [10]. Using a combined CNN+LSTM approach, the model boasts commendable accuracy and F1 score metrics. Specifically, on the UNSW-NB15 dataset, it records accuracies of 93.21% and 92.9% for binary and multi-class classifications respectively. The UNSW-NB15 is an imbalanced dataset, which could lead to overfitting in the model [11]. Meanwhile, on the X-IIoTID dataset, its performance peaks at 99.84% for binary classification and 99.80% for multi-class. Such robust results attest to the model's prowess in intrusion detection for IIoT contexts. Nonetheless, there's room for refining aspects like scalability, general applicability, and addressing certain limitations.

Further, a novel deep anomaly detection system for wind turbines using a Graph Convolutional Autoencoder is presented in [12]. With an impressive F1-score of 91%, the system accentuates the significance of Condition Monitoring tactics. It perceives the sensor network as a fluctuating graph, amalgamating Autoencoders with Graph Convolutional Networks. The data encompasses readings from four turbines spanning 20 months, meticulously sieving out erroneous alerts. This approach may not generalize well to other types of industrial systems due to the specific nature of wind turbine data.

The DCNN algorithm, applied to examine network traffic and identify unauthorized actions, shows remarkable accuracy (between 99.79% and 100%) across multiple IDS datasets [13]. Employing GPU acceleration for improved efficiency, this technique parallels our study in deploying sophisticated deep learning for dependable IDS in SCADA systems. Nonetheless, the uniformity in all evaluation metrics, specifically achieving 99.96% across Accuracy, Precision, Recall, and F1-Score, suggests potential overfitting in the model.

Self-similarity is a concept that has been increasingly recognized and valued in time series analysis. It denotes the consistent behavior of a signal's statistical attributes even when subjected to rescaling transformations. Such a trait becomes particularly advantageous when analyzing data patterns that repeat at different scales, often found in SCADA systems [14].

The Hurst parameter, also known as the Hurst exponent, emerges as a crucial statistical tool in this scenario. It measures the self-similarity degree in a time series, providing insights into its long-term memory or persistence [15]. When utilized in SCADA's Intrusion Detection Systems (IDS), the Hurst parameter aids in identifying the system's inherent patterns and behaviors, facilitating a deeper comprehension of its standard operational stance. This enhanced understanding is vital for spotting anomalies or deviations that could signify potential security risks or system malfunctions. The following sections will explore various works on self-similarity and anomaly detection, summarized in Table 1.

Other studies like [16] utilize cosine similarity for anomaly detection within vehicular networks, a domain that's still burgeoning. Likewise, [17] introduces an innovative approach called dynamic evolving Cauchy possibilistic clustering rooted in the principle of self-similarity (DECS) for crafting an Intrusion Detection System. Yet, a comprehensive evaluation of the reliability and precision of these methodologies remains essential, especially when juxtaposed against other established techniques in the field. While the KDD99 and UNSW-NB15 datasets used in these studies provide valuable insights, their ability to mimic real-world situations can

TABLE 1. Summary of existing studies utilizing self-similarity and machine learning techniques for intrusion detection.

Work	Year	Domain	Technique	Dataset
[7]	2020	IoT Networks	LSTM for DDoS detection	Reflection-based (DrDoS)
[8]	2020	IoT Networks	SMO and SDPN for sensor data detection	NSL-KDD
[8]	2020	Web Protocols	CNN for HTTP intrusion detection	CICIDS2017
[10]	2023	IIoT Networks	CNN + LSTM	UNSW-NB15, X-IIoTID
[12]	2022	Wind Turbines	Graph Convolutional Autoencoder	Wind farm 1 - failures
[13]	2023	NA	DCNN	Multiple IDS datasets
[14]	2016	SCADA Systems	Self-similarity	KDD99
[15]	2015	Simulated Network	Self-similarity (Hurst Parameter)	NT1-HU, NT2-HUF
[16]	2021	Vehicular Networks	Self-similarity (Cosine similarity)	Extracted from two real vehicles in driving and stationary conditions
[17]	2022	NA	DECS	KDD99, UNSW-NB15
[18]	2020	Smart Power Grids	Self-similarity	NA
[19]	2016	LAN and WAN	Self-similarity (Hurst Parameter)	CADIA 2007, MIT normal traffic
[20]	2020	IoT and WSMN	Self-similarity (Hurst Parameter)	Testbed Simulation

be debated, potentially affecting the universality of the conclusions drawn.

In related work, the paper by [18] explores an innovative approach for detecting cyber attacks on smart power supply networks by analyzing the self-similarity property of network traffic. The methodology examines long-term dependencies in fractal Brownian motion and real network traffic of smart grid systems. This approach aids in the identification of anomalies in network traffic which could be indicative of cyber attacks. It manifests the self-similarity in smart grid systems' network traffic and underscores the potential of this approach in swiftly detecting anomalies.

A method for detecting Distributed Denial-of-Service (DDoS) attacks is explored by leveraging the concept of self-similarity in network traffic, as detailed in [19]. Recognizing the resemblance between high-rate attack traffic, low-rate attack traffic, and legitimate traffic is crucial as self-similarity exists in ethernet traffic, which can be measured using the Hurst parameter. This parameter is used to gauge the local irregularity or self-similarity of traffic under a DDoS flood attack, given that fractional Gaussian noise (fGn) is employed as the traffic model.

Song et al. in [20], primarily focuses on the IoT and the Wireless Sensor Multimedia Networks (WSMN) security, particularly in detecting and preventing network anomalies. The technique deployed revolves around a method of forming a set of informative features based on the evaluation of the Hurst (H) parameter of network traffic and the Three Sigma Rule, alongside the development of algorithms for a new Deep Packet Inspection (DPI) system that can analyze captured traffic, detect protocols, and determine statistical load parameters. Real-time analysis is vital for the IoT, but this method might struggle with achieving real-time performance due to the extensive computational demands of deep packet inspection.

In this research, we introduce an enhanced hybrid model designed for anomaly detection within SCADA systems. By integrating the insights from the self-similarity characterized by the Hurst parameter with the modified CNN-LSTM model and incorporating PCA for dimensionality reduction, our approach aims for optimized performance.

For a robust training and evaluation phase, we employed the CICIDS2017 dataset, a detailed collection of network traffic traces curated by the Canadian Institute for Cybersecurity. This dataset provides an authentic depiction of modern network interactions, presenting both benign and malicious traffic patterns. Our selection of this dataset is also influenced by its ability to surpass the limitations found in older datasets, rendering it ideal for model validation.

Diverging from conventional methodologies, this research carves a novel path by meticulously integrating the Self-similarity Hurst parameter as an augmented feature into the CICIDS2017 dataset, while developing an IDS with a CNN-LSTM model. Unlike previous studies, the novelty here lies not just in utilizing the Hurst parameter, but ingeniously embedding it as an additional feature, thereby enriching the dataset and enhancing the model's capability to discern and predict anomalies and potential intrusions with remarkable accuracy and reliability.

This strategic integration of self-similarity, encapsulated by the Hurst parameter, into the CICIDS2017 dataset, married with a modified CNN-LSTM model, unveils a pioneering methodology that is both theoretically robust and practically impactful, providing a fortified defense mechanism against evolving cyber threats in SCADA systems.

The ensuing sections delve deeper into the methodology, implementation, and notable results derived from this innovative approach, illuminating its potential to act as a linchpin in safeguarding SCADA system security amidst an ever-complex cybersecurity landscape.

III. METHODOLOGY

This research seeks to evaluate the effectiveness of integrating the Self-similarity Hurst parameter with the CNN-LSTM in detecting anomalies within SCADA systems. Our approach encompassed a series of steps, starting with data preparation, followed by Hurst parameter computation, feature selection via PCA, training the combined model, and ultimately,

assessing its performance. This section lays out the methods used to answer the research question. To provide a clear overview of the process, a pseudo algorithm summarizing these steps is presented below.

Algorithm 1 Proposed Methodology for SCADA System Anomaly Detection

Input: CICIDS2017 dataset
Output: Anomaly detection model performance metrics
1: Load the CICIDS2017 dataset
2: Perform data cleaning
3: Conduct statistical analysis and data visualization
4: Use Random Forest classifier for feature importance determination
5: Apply PCA to reduce dimensionality, aiming for 95% explained variance
6: Divide the data into segments
7: Calculate R and S for each
8: Calculate the value of log(R/S) ratio
9: Include batch normalization and dropout layers for optimization
10: Combine features from PCA and Hurst parameter for model input
11: Train the model on the preprocessed dataset
12: Evaluate the model using accuracy, recall, precision, and F1-score
13: Visualize the results
14: Analyze the model's architecture and performance

A. DATASET DESCRIPTION

CICIDS2017 is a dataset diligently designed by the Canadian Institute for Cybersecurity (CIC) [6] that offers an extensive collection of network traffic patterns, making it apt for Intrusion Detection Systems (IDS) assessments. Selected for its representation of real-world network activities over a week, this dataset encompasses both harmless and malevolent traffic patterns. It offers an advantage over older datasets, such as KDD99, by including recent attack strategies. CICIDS2017 documents a broad spectrum of cyber attacks, including but not limited to Brute Force, DoS, DDoS, Heartbleed, and Web Attacks. The dataset consists of raw traffic data, packets, and flow-driven datasets that portray both innocuous and harmful traffic in detail. Importantly, it sheds light on aspects of both the network and system, enhancing its value for in-depth studies. In this investigation, we utilized CICIDS2017 to both train and assess the combined CNN-LSTM and Self-Similarity model.

B. CICIDS2017 EXPLORATORY DATA ANALYSIS (EDA)

In our study of the CICIDS2017 dataset, we delved into an in-depth Exploratory Data Analysis (EDA). Beginning with a raw data inspection, the process encompassed exploration, statistical assessments, data refinement, and visualization stages. The final step involved using a Random Forest classifier to gauge feature importance, setting the stage for effective modeling. Figure 1 shows the workflow of the CICIDS2017 EDA.

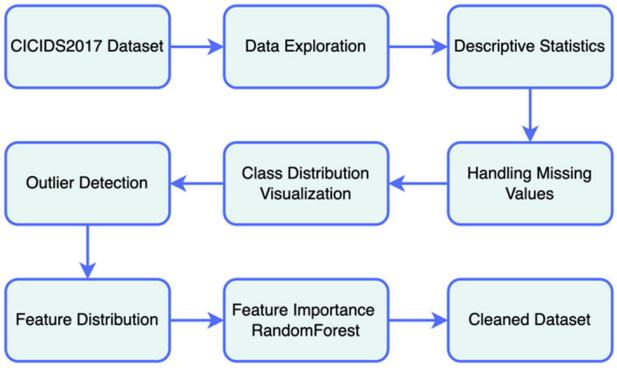

FIGURE 1. This diagram depicts our EDA process for the CICIDS2017 dataset, from raw data to feature importance via a random forest classifier, illuminating the dataset's intricacies vital for our hybrid ids approach.

In the initial phase of data preprocessing, we addressed the presence of missing values within the dataset. Considering the extensive volume of the CICIDS2017 dataset, the elimination of these missing values is not expected to adversely affect the overall quality and reliability of the dataset. To ensure that no missing values were left in the dataset post-cleanup, the data.isnull().sum() function was utilized as a verification tool.

Regarding the normalization process, we opted for the Min-Max scaling technique, primarily due to its straightforward and uncomplicated approach. This method is particularly effective in adjusting the data into a specific range, which is beneficial for maintaining consistency across different features, thereby facilitating more accurate analysis and modeling in our study.

The CICIDS2017 dataset, a significant resource in cybersecurity, offers detailed insights into SCADA system network traffic. Housing over 2.8 million records and 78 unique features, it presents a wealth of information. We noted that the average destination port value is approximately 8071, and nearly half of the records resonate with port 80, indicating a dominance of HTTP traffic. Any deviation here, especially in a SCADA context, might hint at underlying threats. The missing values are removed from the dataset as it's a large dataset and removing a few records won't affect the dataset quality.

The packet dynamics within the dataset revealed interesting patterns. While the average forward packet count sits at 9, there are outliers scaling to the hundreds of thousands. This divergence may be indicative of SCADA system anomalies. Additionally, the packet lengths within the CICIDS2017 dataset showed SCADA communications tend to have longer response packets than the initiating ones.

We also noticed noteworthy patterns related to network activity durations and extended idle periods, potential markers of SCADA system behaviors. Data anomalies, such as negative values in *'min_seg_size_forward'*, deserve special

attention. Moreover, the dataset's right-skewed distribution highlighted potential outliers, essential for SCADA anomaly detection.

For visualization, we leveraged Seaborn, a Python-based tool, to depict class distribution in the *'Label'* column. The visualization in Figure 2 underscored the imbalance in the dataset: the *'Benign'* class -the second highest class in CICIDS2017- has around 2.2 million records, vastly outnumbering the *'DoS Hulk'* class with approximately 230,000 entries.

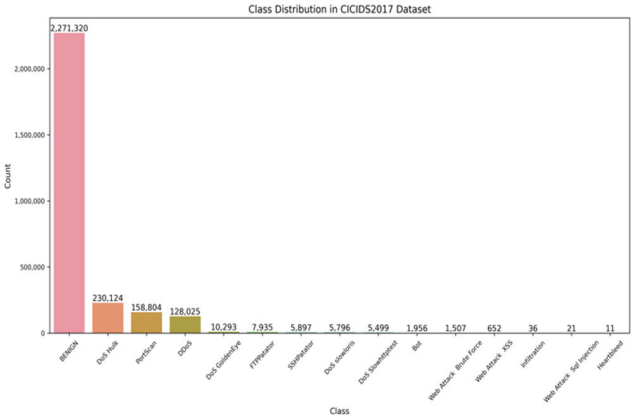

FIGURE 2. Class distribution in the CICIDS2017 dataset with around 78.6% normal records.

This study deliberately does not address the data imbalance aspect in the CICIDS2017 dataset to maintain a specific focus on isolating the contribution of the Self-similarity feature. We had previously worked on the analysis of dataset imbalance in detail in [21]. However, since we cannot predict the effect of balancing the dataset on the performance of the proposed scheme, we certainly intend to study it in the immediate future work.

The Z-score, or standard score, quantifies how far a data point deviates from the mean of the dataset in terms of standard deviations, and is commonly visualized using a normal distribution curve. This metric is particularly valuable in deep learning for outlier detection, as it adeptly measures deviation in datasets characterized by numerous dimensions or features [22].

In our study, we utilized the Z-score methodology to pinpoint and exclude outliers. These outliers represent data points that significantly diverge from the norm, potentially leading to skewed analytical outcomes. By eliminating these outliers, we enhance the model's precision, minimizing the influence of extreme values that might not accurately represent typical network traffic patterns. This is vital for the model's effective recognition of patterns that signify cyber threats. Nevertheless, we exercised caution to avoid excessive data removal, preserving the dataset's integrity and its relevance to real-world scenarios. Such careful management of outliers fortifies the robustness and reliability of our research.

For accurate outlier detection, we enlisted the Z-score method. This statistical tool, proficient in pinpointing significant mean deviations, offers an objective lens for anomaly detection.

$$Z = \frac{\chi - \mu}{\sigma} \tag{1}$$

where Z is the z-score, χ is the observation value, μ is the data mean, and σ is the data standard deviation. Applying this method, we spotlighted outliers within the features, excluding *'Label'*. In the realm of cybersecurity, outliers can either signify potential threats or be mere data noise. It's essential to differentiate between the two.

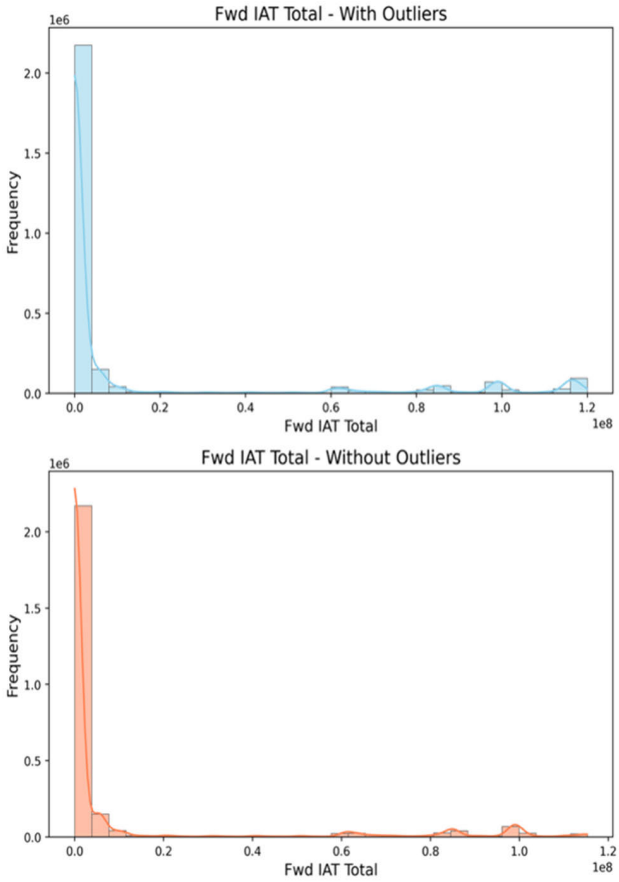

FIGURE 3. Histograms of 'Fwd inter-arrival time (IAT) total' with and without the outliers.

We observed significant outliers in the *'Fwd IAT Total'* and *'Bwd IAT Total'* features, informing our next steps in the modeling process. Figure 3 displays the histograms of the 'Fwd IAT Total' with and without outliers. Similarly, Figure 4 shows the same for the 'Bwd IAT Total'.

Feature importance is a crucial aspect of machine learning and data analytics that helps in determining the significance of each feature in predictive modeling. Essentially, it quantifies the impact of each feature on the model's outcome, enabling us to prioritize and understand the data better.

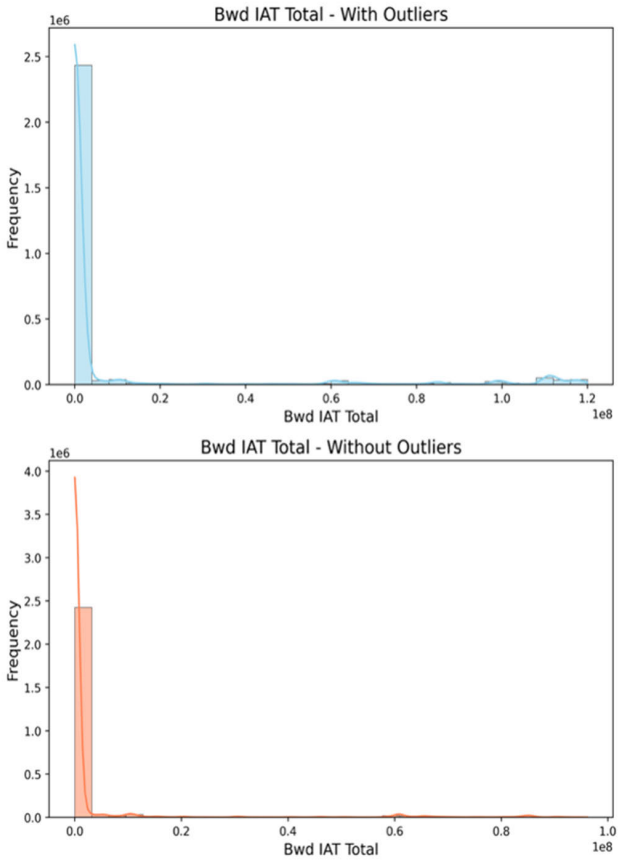

FIGURE 4. Histograms of 'Bwd IAT total' with and without the outliers.

In the context of CICIDS2017, a dataset related to cybersecurity, feature importance can be pivotal in discerning which attributes play a significant role in identifying potential threats or anomalies.

The Random Forest algorithm, a popular ensemble method, evaluates feature importance by observing how often a feature is utilized to split the data and how much it improves the purity of the split. This mechanism allows Random Forest to provide a rank order of features based on their contribution to the model's predictive accuracy.

In our analysis of the CICIDS2017 dataset, using Random Forest enabled us to pinpoint the top 10 features shown in Figure 5, thereby offering valuable insights into the most critical parameters to monitor and analyze for cybersecurity purposes.

In summary, our EDA of the CICIDS2017 dataset emphasizes the importance of rigorous data inspection, especially in SCADA systems. Discerning these data patterns and anomalies can be the difference between a secure and a compromised SCADA environment.

C. DIMENSIONALITY REDUCTION WITH PCA

To enhance computational efficiency and model performance, we applied Principal Component Analysis (PCA) to reduce the dataset dimensionality, which initially contained

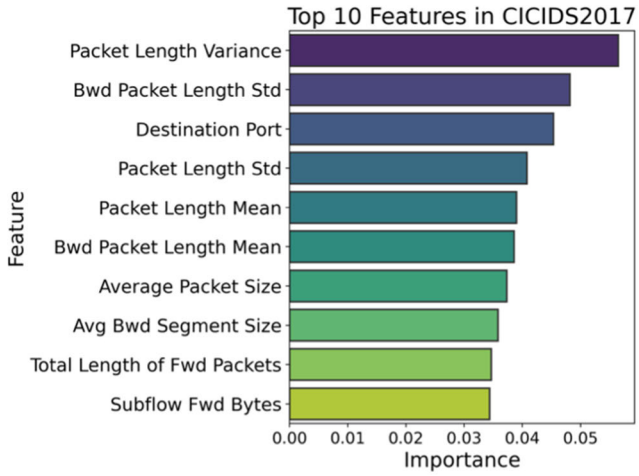

FIGURE 5. This chart highlights the top 10 features in the CICIDS2017 dataset based on their importance in predicting the target variable.

78 features. Our goal was to capture 95% of the dataset's variance, thereby retaining the most significant features for network behavior analysis.

PCA transforms the original set of features into new, uncorrelated variables, known as principal components. These components are linear combinations of the original variables, selected based on their contribution to the total variance in the dataset [23]. The process of selecting these components involved identifying the ones that cumulatively explained 95% of the variance, ensuring the inclusion of the most informative features for intrusion detection while discarding redundant data. The retained principal components are representations of underlying data structures that are crucial for identifying potential security threats in SCADA systems.

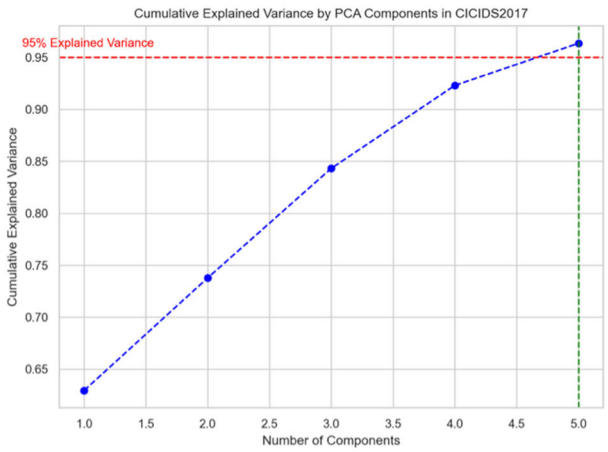

FIGURE 6. The plot illustrates the proportion of total variance captured as more components are considered. The red dashed line signifies the 95% explained variance threshold, while the green dashed line highlights the specific number of components required to achieve this threshold.

The Cumulative Explained Variance by PCA Components graph, depicted in Figure 6, displays the cumulative

variance explained by the initial n principal components. This visualization was pivotal in our study, allowing us to determine the optimal number of components necessary for a balance between preserving critical information and achieving dimensionality reduction. This balance is essential for computational efficiency and ensuring model clarity in detecting anomalies within SCADA systems.

D. HURST PARAMETER

The Hurst parameter is a statistical measure used to assess the self-similarity of a signal or pattern, often in the context of time series data [24]. It's instrumental in our research for detecting anomalies within the SCADA systems' network traffic, using the CICIDS2017 dataset. We calculate the Hurst parameter for each data record to determine the degree of self-similarity in the network traffic.

Calculation Process:

1. We start by dividing the time series data into equal-length segments. For instance, if a dataset has 100 data points, we might break it into segments containing 10 points each.
2. For each segment, we calculate its range R (the difference between the highest and lowest values) and its standard deviation S (a measure of variation from the mean).
3. We compute the R/S ratio for each segment by dividing its range by its standard deviation.
4. We then take the logarithm of these R/S ratios to normalize the data.
5. The average of these logarithmic values is calculated across all segments.
6. The Hurst parameter is determined using the formula in (2)

$$H = \frac{E[log(\frac{R}{S})]}{log(n)} \tag{2}$$

where H represents the Hurst parameter, R represents the range of the data, S is the standard deviation of the signal, and n represents the time scale.

Imagine a hypothetical dataset where, after performing the above calculations, the average logarithmic R/S value for segments of 10 data points is 0.3. The Hurst parameter, H, would then be calculated as $(0.3/\log(10)) = 0.13$. This value of H helps us understand the degree of self-similarity in the dataset, which is critical for identifying patterns and anomalies in network traffic.

The Hurst parameter plays a vital role in our research, enabling us to identify deviations from typical traffic patterns in the SCADA systems. By computing this parameter for the CICIDS2017 dataset, we can effectively pinpoint unusual activities, which may indicate security threats or system vulnerabilities.

E. BATCH NORMALIZATION

Batch Normalization (BN) is a technique that improves the training of deep learning models by addressing internal covariate shifts and stabilizing the learning process [25]. It normalizes the activations within each layer, reducing the impact of changing input distributions during training. BN also acts as a form of regularization, mitigating the vanishing and exploding gradient problems. Overall, BN enhances training efficiency, improves model performance, and ensures a more stable and effective learning process.

F. DROPOUT LAYERS

Dropout is a regularization method often used in deep learning models to minimize overfitting [26]. Dropout randomly deactivates a portion of neurons in a layer with a predetermined probability during training, essentially "dropping out" their contribution to following layers for a certain forward and backward pass. This process introduces instability into the network while also ensuring that no one neuron gets overly specialized, resulting in a more generalized and robust model. Overall, including dropout layers is a simple yet effective way to improve the generalization capabilities of deep learning models.

G. EVALUATION METRICS

The performance of the proposed hybrid intrusion detection system is evaluated using a set of commonly used evaluation metrics, including accuracy (ACC), recall, precision, and F1-score [27]. These metrics provide a comprehensive evaluation of the system's ability to accurately detect anomalies in the CICIDS2017 dataset.

Accuracy is the most widely used performance metric for binary and multi-class classification problems. In the context of intrusion detection, accuracy measures how accurately the system classifies normal and abnormal network traffic [28]. Recall (also known as True Positive Rate or TPR) indicates the proportion of network anomalies that are correctly identified by the system, while precision measures the proportion of correctly classified positive cases to the total number of positive cases. The F1-score, which is the weighted harmonic average of precision and recall, is particularly useful in classification problems.

H. MODEL ARCHITECTURE

The proposed methodology, seen in Figure 7, presents an improved technique for anomaly detection in SCADA systems by leveraging the invaluable CICIDS2017 dataset. Realizing the crucial significance of SCADA systems in critical infrastructure, there is an evident need for effective security solutions, particularly given their vulnerability to a wide range of cyber-attacks.

A well-constructed neural network is at the core of our research. This network is distinguished by its seamless combination of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) units, resulting in an architecture precisely designed to detect both spatial and temporal patterns in input sequences. This hybrid design not

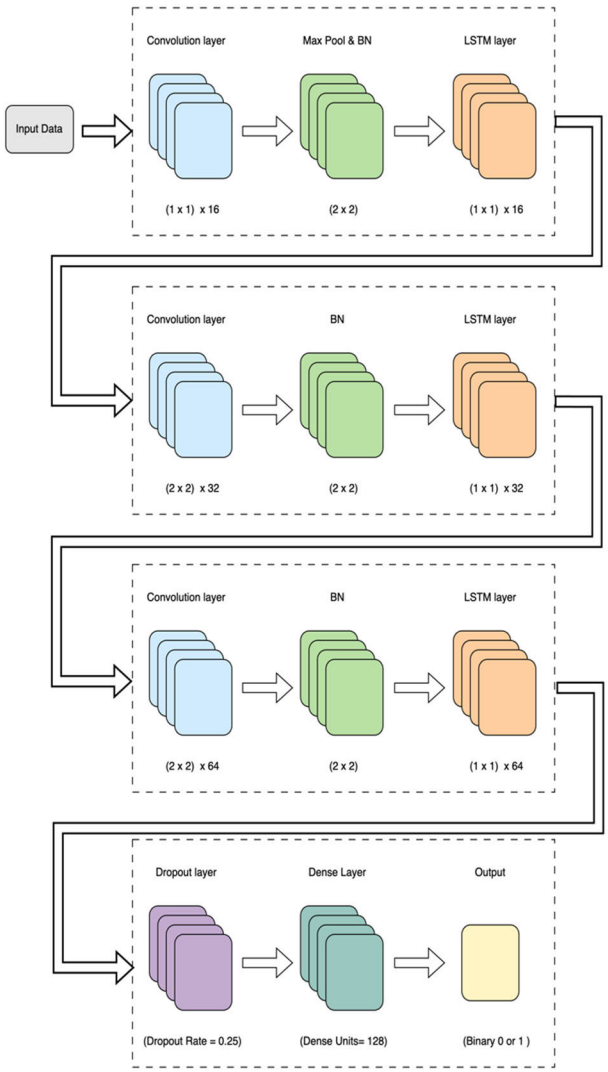

FIGURE 7. The architecture of the hybrid CNN-LSTM neural network model for SCADA system anomaly detection, leveraging the CICIDS2017 dataset.

only provides a deep comprehension of the data but also highlights the complexities of network traffic patterns.

The data's journey begins with a series of preprocessing steps that handle outliers, normalize the data, and transition through PCA. The Hurst parameter, a standout feature, provides profound insights into the time series' behavior and is smoothly included in our feature set, which is constructed with seven different features: six principal components and the Hurst parameter, represented as (7, 1).

The *"ReLU"* activation function is used by our model's CNN layers, which are equipped with increasing filters (16, 32, 64) and kernel sizes of 1 and 2. This decision is based on *ReLU's* ability to overcome the vanishing gradient dilemma and accelerate deep network convergence. Batch normalization layers are distributed throughout this convolutional architecture, critical pieces ensuring stable and accelerated training by limiting any one input feature's overwhelming dominance.

In addition to the convolutional dynamics, our model has three LSTM layers that are gradually organized with 16, 32, and 64 recurrent units. These LSTMs, known for their ability to retain long-term sequential data, are critical for capturing temporal patterns in our dataset.

The model converges the gathered features after the LSTM layers with a dense layer packed with 128 units. This complicated architecture results in a single-neuron output layer primed with a sigmoid activation function, ideal for binary classification applications.

Upon completing the training phase, the model's mettle is gauged through performance metrics like accuracy, precision, recall, and F1 score. By capitalizing on deep learning's immense potential and the comprehensive insights from CICIDS2017, our model heralds a new era in fortifying SCADA system security, ensuring operations that are both secure and steadfast.

Our neural network architecture was influenced by previous work [29], but it diverges significantly in the implementation details and the unique integration of the Hurst parameter. The configuration of layers, units, and activation functions was established through rigorous experimentation, tailored specifically to the CICIDS2017 dataset to optimize anomaly detection in SCADA
systems.

We chose the ReLU activation function for its effectiveness in deep learning models and structured the LSTM layers to incrementally increase from 16 to 64 units based on their performance in capturing temporal data patterns. The incorporation of batch normalization ensures stable training and mitigates common neural network issues. This strategic combination of features and layers results in a robust architecture, finely tuned to address the complexities of network traffic in SCADA systems.

The training process for this SCADA system anomaly detection model involves using a Sequential neural network, composed of several layers including Conv1D, MaxPooling1D, LSTM, Dense, and Dropout. This combination effectively captures both spatial and temporal patterns in the data. The details of the training phase and the hyperparameters are shown in TABLE 2. The model employs ReLU activation for the Conv1D layers to introduce non-linearity and a Sigmoid function in the output layer for binary classification. It utilizes the Adam optimizer for efficient computation and adaptive learning rates, paired with binary cross-entropy as the loss function to measure performance. Key hyperparameters include a batch size of 128, ensuring a balance between computational efficiency and generalization, and the model is trained over 10 epochs. Data preprocessing includes outlier handling, normalization, and Principal Component Analysis (PCA) to reduce dimensionality. Notably, the model incorporates the Hurst parameter, alongside dropout layers to prevent overfitting. This configuration is carefully designed to optimize the model's ability to detect anomalies in SCADA systems.

TABLE 2. Overview of model training process and hyperparameters for SCADA system anomaly detection.

Aspect	Description	Reasoning
Model Type	Sequential	Enables the stacking of layers linearly and provides a straightforward architecture for the neural network.
Layers	- Conv1D (16, 32, 64 filters, kernel sizes 1 & 2) - MaxPooling1D - LSTM (16, 32, 64 units) - Dense (128 units) - Dropout (0.25 rate)	Conv1D and LSTM layers to capture spatial and temporal patterns. Dense and Dropout layers for decision-making and regularization.
Activation Functions	- Conv1D: ReLU - Output layer: Sigmoid	ReLU enhances non-linearity without affecting receptive fields. Sigmoid is suitable for binary classification.
Optimizer	Adam	Provides efficient computation and adaptive learning rates, enhancing convergence.
Learning Rate	0.001	This value represents a balance between convergence speed and stability in a wide range of optimization problems in machine learning.
Loss Function	Binary Crossentropy	Suitable for binary classification tasks, measures the performance of the model.
Metrics	Accuracy, Precision, Recall, F1 Score	Provide a comprehensive evaluation of the model's performance from various aspects.
Input Shape	(7, 1) - after PCA and Hurst parameter augmentation	Reflects the dimensionality of the processed dataset including PCA and Hurst parameter.
Batch Size	128	Balances computational efficiency and the model's ability to generalize across data samples.
Epochs	10	Determines the number of times the entire dataset is passed forward and backward through the network.
Validation Split	15% of training data	Allocates a portion of training data for model validation without using the test set.
Data Preprocessing	- Outlier detection and handling - Normalization - PCA for dimensionality reduction - Hurst parameter computation	Ensures data quality, standardizes feature scales, reduces dimensionality, and incorporates a novel feature (Hurst).
PCA Components	6	Retains most of the variance in data while reducing dimensions to make computations more efficient.
Dropout Rate	0.25	Helps prevent overfitting by randomly deactivating a portion of neurons during training.

IV. RESULTS AND ANALYSIS

In the quest to enhance SCADA security through a hybrid deep learning IDS leveraging self-similarity, a comprehensive series of experiments and analyses were conducted. The results obtained from these activities are detailed in this section.

A. PREPROCESSING RESULTS

Upon our initial exploration of the dataset, we identified a total of 2,830,743 samples with 79 features in tow. The benign samples, comprising approximately 80.35% (or 2,273,097 samples), formed a significant portion of the dataset. In stark contrast, the attack samples, which totaled 557,646, accounted for the remaining 19.65%.

During the preprocessing phase, our outlier analysis revealed inconsistencies and anomalies across a majority of the features. As depicted in Figure 8, a staggering 70 features were affected by outliers. This bar chart offers a bird's-eye view of the outlier distribution across each feature, emphasizing the necessity of addressing these irregularities to ensure the robustness and reliability of our model.

With the aim of reducing the dataset's dimensionality, we employed Principal Component Analysis (PCA). Figure 9 illustrates how the initial 5 components capture a significant chunk of the dataset's information. The first principal component stands out, explaining 21.51% of the variance. Implementing this dimensionality reduction not only accelerates the training phase but also promises improved model generalization by sidestepping the pitfalls of the curse of dimensionality.

The progression of the dataset as detailed in Table 3 reveals a meticulous approach to data preparation crucial for reliable machine learning models. Starting with a substantial dataset, preprocessing ensures data quality and consistency. The significant reduction in feature count post-PCA, from 78 to 6, highlights the effectiveness of dimensionality reduction in capturing essential information while reducing computational load. The addition of the Hurst parameter, increasing the feature count to 7, illustrates a strategic enhancement, introducing a nuanced aspect for anomaly detection.

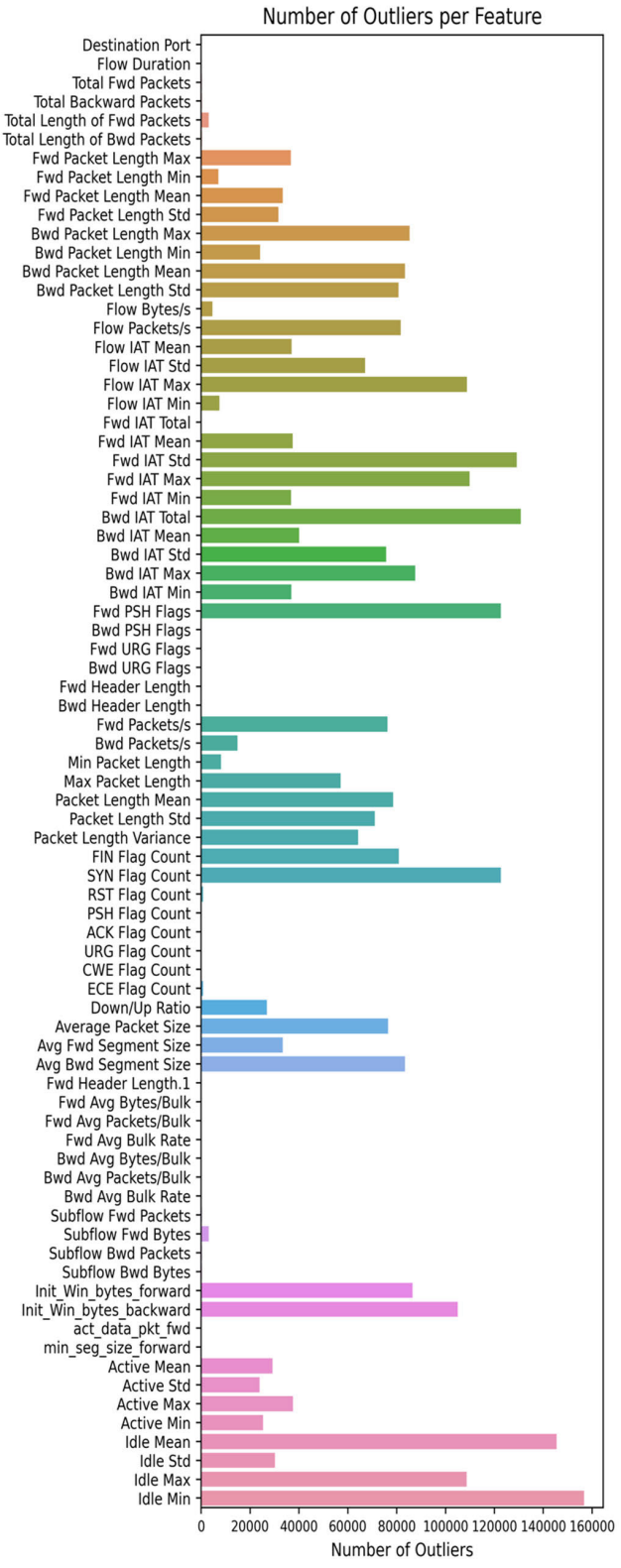

FIGURE 8. The number of outliers detected in each feature.

The consistent sample count throughout these stages ensures robust model training and evaluation. This thorough data preparation process underpins the model's potential for accurate and efficient anomaly detection in SCADA systems.

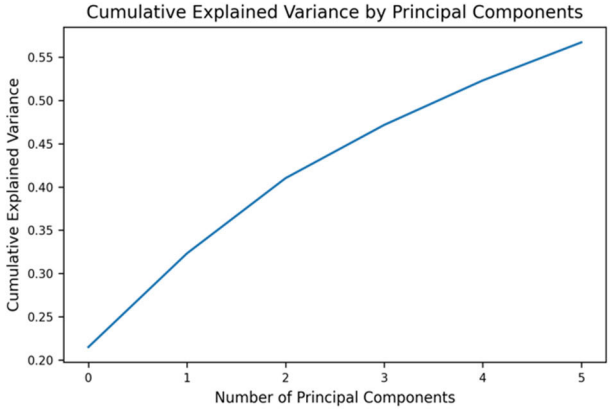

FIGURE 9. Cumulative explained variance by principal components.

TABLE 3. Stages of dataset processing from initial load to feature augmentation.

Stage	Sample Count	Feature Count	Description
Initial Data Load	2,830,743	79	Raw combined data from multiple files.
After Preprocessing (Duplicates Removed)	2,520,798	79	Data cleaned; duplicates removed.
After Outlier Handling	2,520,798	79	Outliers detected in 70 features and handled.
Dataset Shape After Separating X and Y	2,520,798	X: 78, Y: 1	Features (X) and Labels (Y) are separated.
After Train-Test Split (Training Data)	2,016,638	78	Data is split into training and testing sets.
After Train-Test Split (Testing Data)	504,160	78	A separate testing set is preserved for model evaluation.
After PCA	2,016,638	6	Dimensionality is reduced to 6 principal components.
After Hurst Parameter Added	2,016,638	7	Hurst parameter calculated and added as a feature.

B. ANALYSIS OF PCA COMPONENT INTERPRETABILITY

The application of PCA in our study has led to the extraction of principal components (PCs) that encapsulate the most significant features of the network traffic data in the CICIDS 2017 dataset. These components, while reducing the dimensionality of the dataset, retain the essential characteristics that are pivotal for anomaly detection in SCADA systems. For instance, PC1, which explains 21.51%

TABLE 4. Top principal components and their key contributing features.

Principal Component	Explained Variance (%)	Top Contributing Features
PC1	21.51	Flow IAT Max, Fwd IAT Max, act_data_pkt_fwd, Init_Win_bytes_forward
PC2	10.82	Avg Bwd Segment Size, Total Fwd Packets, Total Backward Packets
PC3	8.69	Bwd IAT Mean, Fwd IAT Min, Bwd IAT Min
PC4	6.15	Down/Up Ratio, Fwd Packet Length Mean
PC5	5.14	RST Flag Count, Flow Packets/s, Fwd Header Length
PC6	4.40	Subflow Bwd Packets, Subflow Fwd Packets, Subflow Fwd Bytes

of the variance, is heavily influenced by features like 'Flow IAT Max' and 'Fwd IAT Max', see Table 4. These features are indicative of the maximum inter-arrival times within the network, which is crucial for identifying unusual delays or rapid sequences of data packets — common signs of network anomalies or cyberattacks. Table 4 presents the PC components, the percentage of explained variance, and their most influential features.

Similarly, other PCs highlight different but equally significant aspects of network behavior. For example, PC2 focuses on segment sizes and packet counts, which are vital in recognizing unusual traffic volumes that could signal a Distributed Denial of Service (DDoS) attack. PC3's emphasis on 'Bwd IAT Mean' and 'Fwd IAT Min' further enhances our understanding of the temporal patterns in the data flow, which is essential in detecting advanced persistent threats that may exhibit slow, methodical patterns of behavior over time.

The principal components derived from PCA offer real-world insights into network behaviors that are critical for SCADA systems' security. In practical terms, these components serve as condensed representations of intricate patterns in network traffic, enabling the model to detect anomalies with greater accuracy. For instance, a significant variance in PC1 could indicate irregularities in data transmission intervals, a potential red flag for cybersecurity teams monitoring SCADA systems.

Similarly, the contributions of PCs in highlighting packet sizes and counts (as seen in PC2 and PC3) are instrumental in real-world scenarios. An unexpected increase in packet size or count, as captured by these PCs, could suggest an ongoing cyberattack, prompting immediate investigation and response. This is particularly relevant in industrial contexts where SCADA systems control critical infrastructure, and any deviation from the norm can have significant implications for operational stability and safety.

The PCs' relation to network behavior is evident in their ability to capture and highlight specific traffic patterns that are generally associated with cyber threats. For example, the focus on inter-arrival times and packet lengths aids in identifying potential network scanning activities or the presence of malware that often disrupts normal traffic patterns. The nuanced understanding of these behaviors, facilitated by the PCs, is critical in developing robust intrusion detection systems that can effectively counter the sophisticated and evolving nature of cyber threats in SCADA environments.

In essence, the PCA component analysis not only simplifies the computational demands of the model but also enhances its interpretability and real-world applicability. By distilling complex data into key features that directly correlate with network behaviors indicative of security threats, the PCA components play an indispensable role in the robust anomaly detection capabilities of our proposed model.

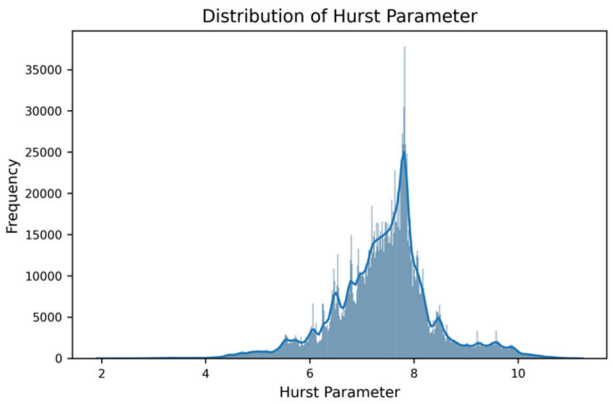

FIGURE 10. Distribution of the Hurst parameter across the dataset.

The Hurst parameter is introduced as a distinctive feature within our dataset to quantify self-similarity in network traffic patterns, which is a critical aspect of modern intrusion detection. Figure 10 illustrates the distribution of the Hurst parameter, highlighting its variance across the dataset. This addition is not merely an augmentation but is central to our contribution; it enhances the model's precision in discerning complex and subtle anomalies, which might escape detection by traditional IDS methods.

By employing the Hurst parameter in conjunction with PCA, our proposed model is not only equipped to detect overt intrusions but also equipped to unravel and flag the more sophisticated, latent patterns that typify advanced network threats. This strategic enhancement is pivotal in bolstering the model's diagnostic acumen, thereby advancing the field of intrusion detection.

C. MODEL PERFORMANCE AND COMPARATIVE ANALYSIS

Our endeavor to comprehend the role of the Hurst parameter in network intrusion detection led us to design two distinct training scenarios: one incorporating the Hurst parameter and the other excluding it. This bifurcation serves as a foundational step in our investigation, enabling a detailed exploration of each training's nuances and providing a platform for a comparative assessment.

The training performance charts serve as a window into the model's learning dynamics. With the Hurst parameter integrated, the convergence pattern as depicted in Figure 11 showcases a harmonious balance between the training and validation phases, underscoring the model's adeptness in generalization.

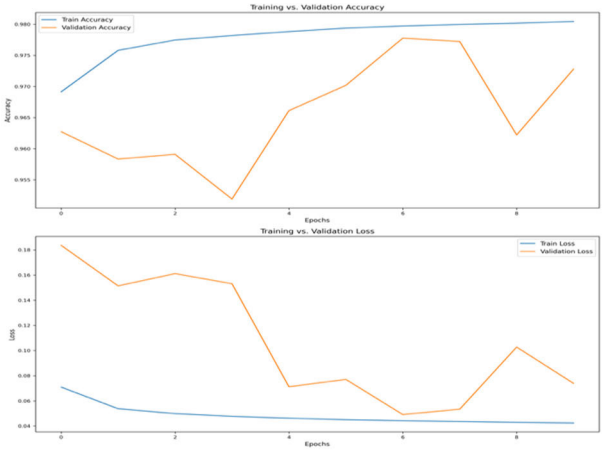

FIGURE 11. Training and validation accuracy and loss metrics across epochs for the model incorporating the Hurst parameter.

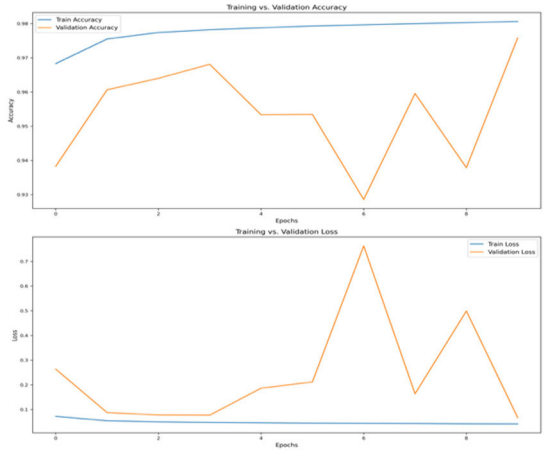

FIGURE 12. Epoch-wise training and validation accuracy and loss metrics for the model excluding the Hurst parameter.

In contrast, the training evolution without the Hurst parameter, visualized in Figure 12, divulges a more capricious journey. While overall convergence is achieved, the validation metrics exhibit occasional turbulence, hinting at a potential overfitting scenario.

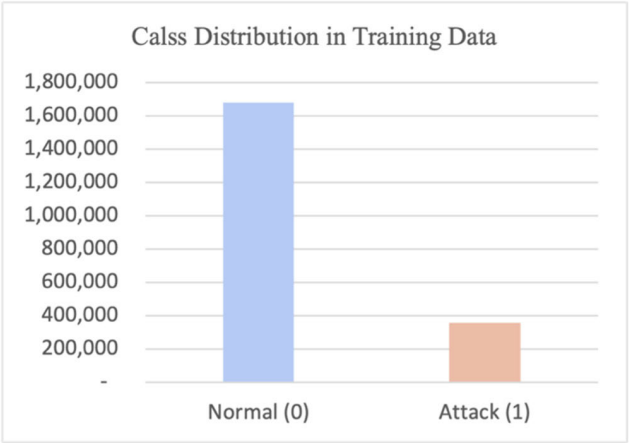

FIGURE 13. Bar chart representation of the class distribution in the training dataset.

The heartbeat of any classification task lies in its data distribution. Figures 13 and 14 unfurl the class distributions for the training and testing datasets, respectively. The equilibrium observed in the training dataset ensures unbiased learning, whereas the test data presents a slight tilt, making recall an invaluable metric in our assessment arsenal.

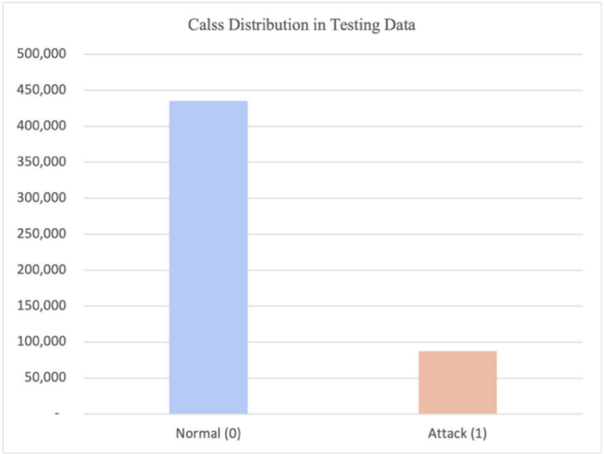

FIGURE 14. Class distribution for the test dataset, visualized as a bar chart.

The confusion matrices, our compass in navigating the landscape of predictions, further accentuate this narrative. Figures 15 and 16 representing the matrices with and without the Hurst parameter respectively, shed light on the model's nuances in classification. A discernible shift in false negatives between the two scenarios emerges, spotlighting the amplification in recall when the Hurst parameter is intertwined. A synthesized view of the performance metrics is encapsulated in Table 5.

False negatives in SCADA system IDS can have severe consequences, including undetected cyberattacks leading to operational disruptions and infrastructure damage, heightened safety risks, and significant economic losses [30].

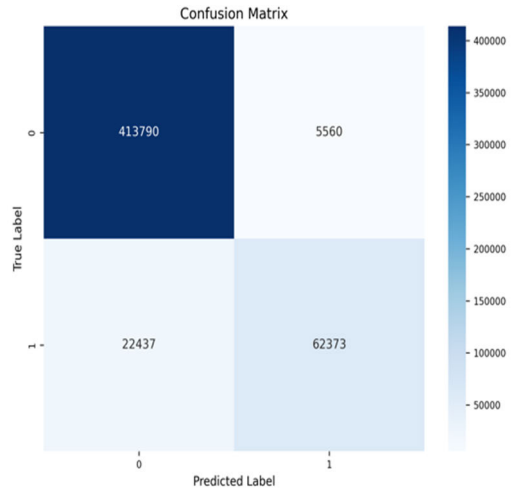

FIGURE 15. Confusion matrix for the model trained with the Hurst parameter.

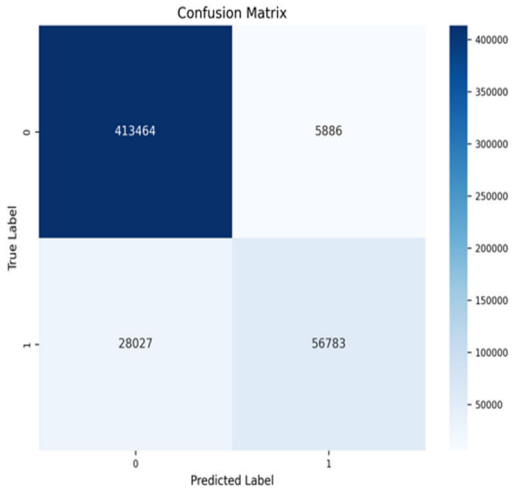

FIGURE 16. Confusion matrix for the model devoid of the Hurst parameter.

TABLE 5. Comparative performance metrics for models with and without the Hurst parameter.

Metric	With Hurst	Without Hurst	P-Value
Accuracy	95.21%	93.65%	0.0047
Precision	88.76%	94.41%	0.0028
Recall	82.59%	65.11%	0.0000
F1 Score	84.14%	74.99%	0.0002

These oversights can undermine public trust, result in non-compliance with regulations, and complicate response efforts. Prolonged undetected intrusions also expose systems to escalated future attacks, exacerbating the risk to critical infrastructure integrity and public safety.

The integration of the Hurst parameter into our SCADA Intrusion Detection System (IDS) represents a significant advancement in its performance metrics, as evidenced both quantitatively and statistically. The Hurst parameter, a tool for detecting long-term trends in time series data, has been instrumental in enhancing the system's accuracy and overall threat detection capabilities.

From a quantitative perspective, the implementation of the Hurst parameter has led to notable improvements in various performance metrics. The accuracy of the system, which is paramount in the realm of intrusion detection, increased from 93.65% to 95.21% with the Hurst parameter. This improvement, while seemingly modest, is substantial in a field where every small increment in accuracy can have significant implications. The recall rate experienced a more pronounced increase, soaring from 65.11% to 82.59%. This substantial jump in recall is particularly important, as it reflects the system's heightened ability to detect actual threats, a critical aspect for IDS in safeguarding sensitive infrastructure.

However, the inclusion of the Hurst parameter did result in a slight decrease in precision, from 94.41% to 88.76%. Despite this, the overall balance between precision and recall, as indicated by the F1 score, improved significantly. The F1 score rose from 74.99% to 84.14% with the inclusion of the Hurst parameter, highlighting its role in achieving a more effective balance between accurately identifying threats and minimizing false alarms.

The decrease in precision can be attributed to the Hurst parameter's increased sensitivity to anomalies, which heightens the system's alertness to potential threats but also results in more false positives.

The statistical significance of these improvements is further substantiated by the p-values obtained through the t-test, as shown in Table 5. These p-values are crucial for determining the likelihood that the observed changes are due to the Hurst parameter rather than random chance. For accuracy, the p-value is 0.0047, for precision, it's 0.0028, for recall, it's remarkably 0.0000, and for the F1 score, it's 0.0002. These low p-values indicate a high level of statistical significance, strongly suggesting that the improvements in the system's performance metrics are directly attributable to the inclusion of the Hurst parameter.

In summary, the integration of the Hurst parameter into the SCADA IDS model has led to statistically significant improvements in key performance metrics. This enhancement is critical for the deployment of an effective IDS within the critical infrastructure of SCADA systems. The improved recall rate and F1 score underscore the value of the Hurst parameter in enhancing the system's ability to detect sophisticated cyber threats, while the statistical analysis confirms the reliability of these improvements. This advancement suggests promising avenues for further enhancing intrusion

detection systems in industrial control systems, ensuring better protection against an evolving landscape of cyber threats.

In assessing the robustness of our model, the architecture's resilience to overfitting is evidenced by the stable performance metrics across training and validation datasets, particularly when the Hurst parameter is incorporated. The confusion matrices (Figures 15 and 16) reveal a consistent decrease in false negatives, indicating the model's enhanced reliability in capturing true anomalies within SCADA systems.

Regarding the complexity analysis, the model's depth and computational layers have been meticulously designed to balance performance with computational demands. This is illustrated by the strategic placement of Dropout layers, which mitigate overfitting while maintaining model complexity at a manageable level. Such architectural decisions underscore the model's suitability for deployment in diverse SCADA environments, offering a robust solution without imposing excessive computational costs.

In this study, it is pertinent to reference our previous work on dataset imbalance [21], elucidating how it informs the current study's methodology. This reference provides a backdrop against which the decision to focus exclusively on the Self-similarity feature's contribution can be understood. Acknowledging this past work also allows for a critical assessment of the limitations of the current study, particularly in terms of dataset representativeness and model generalizability. Moreover, it underscores the importance of future research avenues, including the need for k-fold cross-validation using varied datasets and continuous learning approaches, to enhance the robustness and applicability of our proposed model in the dynamic field of SCADA system cybersecurity.

TABLE 6. Benchmarking based on the cicids2017 dataset.

Author	Algorithm	Accuracy	*Precision*	*Recall*	*F1 Score*
[8]	CNN-LSTM	93.00%	86.74%	76.83%	81.36%
[13]	DCNN	99.96%	99.96%	99.96%	99.96%
Our Work	CNN-LSTM	95.21%	88.76%	82.59%	84.14%

Table 6 presents a comparative analysis of different algorithms benchmarked using the CICIDS2017 dataset. The study by [8] implemented a CNN-LSTM model, achieving 93.00% accuracy, 86.74% precision, 76.83% recall, and an F1 score of 81.36%. In contrast, [13] utilized a DCNN algorithm, achieving exceptionally high and uniform metrics across all categories (99.96% for accuracy, precision, recall, and F1 score), which might indicate overfitting. Our work, also employing a CNN-LSTM model, demonstrated improved performance compared to [8], with 95.21% accuracy, 88.76% precision, 82.59% recall, and an F1 score of 84.14%.

The implementation of a Self-Similarity Deep Learning Hybrid IDS in SCADA systems, as demonstrated in our study, offers substantial real-world benefits. This advanced model significantly improves anomaly detection, allowing SCADA operators to identify and respond to potential threats more accurately and quickly. Its high accuracy and recall rates enable an effective early warning system, crucial for proactive threat mitigation.

The deep learning approach reduces manual monitoring effort, enhancing operational efficiency, and the model's adaptability makes it a robust defense against evolving cyber threats. Customizable to specific SCADA environments, this system also aids in developing targeted cybersecurity policies and training, while ensuring compliance with regulatory standards. Overall, this innovative IDS represents a major step forward in fortifying the security and operational effectiveness of critical SCADA infrastructure.

V. CONCLUSION

This study emphasizes the need for advanced cybersecurity in Supervisory Control and Data Acquisition (SCADA) systems, crucial for industrial operations. We successfully integrated the Self-similarity Hurst parameter with a CNN-LSTM model, significantly enhancing anomaly detection in SCADA systems. Utilizing the CICIDS2017 dataset and Principal Component Analysis, our hybrid model achieved a high detection accuracy of 95.21% and a recall rate of 82.59%.

Our innovative approach, incorporating the Self-similarity Hurst parameter into the IDS, offers new insights into SCADA cybersecurity and improves IDS accuracy. Future research will extend this model to diverse datasets and operational contexts, applying k-folds and cross-validation techniques for robust evaluation. We also plan to integrate continuous learning methods for better adaptability in dynamic.

The findings significantly contribute to SCADA system security, ensuring the reliability of critical services like electricity and water. The deep learning techniques developed have potential applications in cybersecurity, informing policy formulation for protecting industrial systems.

Despite challenges, such as dataset limitations and model integration complexities, this study lays a foundation for future advancements in SCADA cybersecurity, addressing key areas like computational efficiency, real-time performance, and system integration.

ACKNOWLEDGMENT

The authors would like thank to the IIUM Tuition Fee Waiver program for sponsoring the tuition fees of Assad Balla.

REFERENCES

[1] A. Wali, "Analysis of security challenges in cloud-based SCADA systems: A survey," Tech. Rep., Jul. 2022, doi: 10.36227/techrxiv.20291835.v1.

[2] L. O. Aghenta and M. T. Iqbal, "Low-cost, open source IoT-based SCADA system design using Thinger.IO and ESP32 thing," *Electronics*, vol. 8, no. 8, p. 822, Jul. 2019, doi: 10.3390/electronics8080822.

[3] O. Rabie, P. Balachandran, M. Khojah, and S. Selvarajan, "A proficient ZESO-DRKFC model for smart grid SCADA security," *Electronics*, vol. 11, no. 24, p. 4144, Dec. 2022, doi: 10.3390/electronics11244144.

[4] M. Bhavsar, K. Roy, J. Kelly, and O. Olusola, "Anomaly-based intrusion detection system for IoT application," *Discover Internet Things*, vol. 3, no. 1, p. 5, May 2023, doi: 10.1007/s43926-023-00034-5.

[5] F. Laghrissi, S. Douzi, K. Douzi, and B. Hssina, "Intrusion detection systems using long short-term memory (LSTM)," *J. Big Data*, vol. 8, no. 1, p. 65, Dec. 2021, doi: 10.1186/s40537-021-00448-4.

[6] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, 2018, pp. 108–116.

[7] M. Shurman, R. Khrais, and A. Yateem, "DoS and DDoS attack detection using deep learning and IDS," *Int. Arab J. Inf. Technol.*, vol. 17, no. 4, pp. 655–661, Jul. 2020.

[8] A. Kim, M. Park, and D. H. Lee, "AI-IDS: Application of deep learning to real-time web intrusion detection," *IEEE Access*, vol. 8, pp. 70245–70261, 2020.

[9] S. Ethala and A. Kumarappan, "A hybrid spider monkey and hierarchical particle swarm optimization approach for intrusion detection on Internet of Things," *Sensors*, vol. 22, no. 21, p. 8566, Nov. 2022, doi: 10.3390/s22218566.

[10] H. C. Altunay and Z. Albayrak, "A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks," *Eng. Sci. Technol., Int. J.*, vol. 38, Feb. 2023, Art. no. 101322, doi: 10.1016/j.jestch.2022.101322.

[11] A. B. Babiker, M. H. Habaebi, S. Mubarak, and M. R. Islam, "A detailed analysis of public industrial control system datasets," *Int. J. Secur. Netw.*, vol. 18, no. 4, pp. 245–263, 2023.

[12] E. S. Miele, F. Bonacina, and A. Corsini, "Deep anomaly detection in horizontal axis wind turbines using graph convolutional autoencoders for multivariate time series," *Energy AI*, vol. 8, May 2022, Art. no. 100145, doi: 10.1016/j.egyai.2022.100145.

[13] V. Hnamte and J. Hussain, "Dependable intrusion detection system using deep convolutional neural network: A novel framework and performance evaluation approach," *Telematics Informat. Rep.*, vol. 11, Sep. 2023, Art. no. 100077, doi: 10.1016/j.teler.2023.100077.

[14] S. J. Yu, P. Koh, H. Kwon, D. S. Kim, and H. K. Kim, "Hurst parameter based anomaly detection for intrusion detection system," in *Proc. IEEE Int. Conf. Comput. Inf. Technol. (CIT)*, Dec. 2016, pp. 234–240, doi: 10.1109/CIT.2016.98.

[15] D. A. B. Fernandes, M. Neto, L. F. B. Soares, M. M. Freire, and P. R. M. Inacio, "On the self-similarity of traffic generated by network traffic simulators," in *Modeling and Simulation of Computer Networks and Systems*. U.K.: Elsevier, 2015, pp. 285–311, doi: 10.1016/b978-0-12-800887-4.00010-9.

[16] B. I. Kwak, M. L. Han, and H. K. Kim, "Cosine similarity based anomaly detection methodology for the CAN bus," *Exp. Syst. Appl.*, vol. 166, Mar. 2021, Art. no. 114066.

[17] A. H. Alsaeedi, A. Alfoudi, S. Manickam, R. R. Nuiaa, and M. I. Dohan, "Dynamic evolving Cauchy possibilistic clustering based on the self-similarity principle (DECS) for enhancing intrusion detection system," *Int. J. Intell. Eng. Syst.*, vol. 15, no. 5, pp. 252–260, 2022.

[18] I. Kotenko, I. Saenko, O. Lauta, and A. Kribel, "An approach to detecting cyber attacks against smart power grids based on the analysis of network traffic self-similarity," *Energies*, vol. 13, no. 19, p. 5031, Sep. 2020, doi: 10.3390/en13195031.

[19] R. K. Deka and D. K. Bhattacharyya, "Self-similarity based DDoS attack detection using Hurst parameter," *Secur. Commun. Netw.*, vol. 9, no. 17, pp. 4468–4481, Nov. 2016, doi: 10.1002/sec.1639.

[20] W. Song, M. Beshley, K. Przystupa, H. Beshley, O. Kochan, A. Pryslupskyi, D. Pieniak, and J. Su, "A software deep packet inspection system for network traffic analysis and anomaly detection," *Sensors*, vol. 20, no. 6, p. 1637, Mar. 2020, doi: 10.3390/s20061637.

[21] A. Balla, M. H. Habaebi, E. A. A. Elsheikh, M. R. Islam, and F. M. Suliman, "The effect of dataset imbalance on the performance of SCADA intrusion detection systems," *Sensors*, vol. 23, no. 2, p. 758, Jan. 2023, doi: 10.3390/s23020758.

[22] M. A. Mondal and Z. Rehena, "Road traffic outlier detection technique based on linear regression," *Proc. Comput. Sci.*, vol. 171, pp. 2547–2555, Jan. 2020, doi: 10.1016/j.procs.2020.04.276.

[23] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, R. Kaluri, D. S. Rajput, G. Srivastava, and T. Baker, "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020, doi: 10.1109/ACCESS.2020.2980942.

[24] D. Tang, Y. Feng, S. Zhang, and Z. Qin, "FR-RED: Fractal residual based real-time detection of the LDoS attack," *IEEE Trans. Rel.*, vol. 70, no. 3, pp. 1143–1157, Sep. 2021, doi: 10.1109/TR.2020.3023257.

[25] K. Ramli, N. Hayati, E. Ihsanto, T. S. Gunawan, and A. H. Halbouni, "Development of intrusion detection system using residual feedforward neural network algorithm," in *Proc. 4th Int. Seminar Res. Inf. Technol. Intell. Syst. (ISRITI)*, Dec. 2021, pp. 539–543, doi: 10.1109/ISRITI54043.2021.9702773.

[26] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, pp. 1–74, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[27] D. M. Powers, "Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation," *J. Mach. Learn. Technol.*, 2008. [Online]. Available: https://bioinfopublication.org/pages/journal.php?id=BPJ0000274

[28] Z. Wang, W. Xie, B. Wang, J. Tao, and E. Wang, "A survey on recent advanced research of CPS security," *Appl. Sci.*, vol. 11, no. 9, p. 3751, Apr. 2021.

[29] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022, doi: 10.1109/ACCESS.2022.3206425.

[30] V. K. Singh, H. Ebrahem, and M. Govindarasu, "Security evaluation of two intrusion detection systems in smart grid SCADA environment," in *Proc. North Amer. Power Symp. (NAPS)*, Fargo, ND, USA, Sep. 2018, pp. 1–6, doi: 10.1109/NAPS.2018.8600548.

ASAAD BALLA received the bachelor's degree in computer engineering from International Islamic University Malaysia, in 2020, where he is currently pursuing the M.Sc. degree in engineering with the Department of Electrical and Computer Engineering, Faculty of Engineering. His research interests include software engineering, computer network security, the Internet of Things (IoT), and artificial intelligence (AI).

MOHAMED HADI HABAEBI (Senior Member, IEEE) received the degree from the Civil Aviation and Meteorology High Institute, Libya, in 1991, the M.Sc. degree in electrical engineering from Universiti Teknologi Malaysia, in 1994, and the Ph.D. degree in computer and communication system engineering from University Putra Malaysia, in 2001. Currently, he is a Professor with the Department of Electrical and Computer Engineering, International Islamic University Malaysia. His current research interests include wireless communication protocols, the Internet of Things, wireless sensor and actuator networks, cognitive radio, small antenna systems and radio propagation, and artificial intelligence. He also serves as an active reviewer and he is on the editorial board of several international journals. He is a C.Eng. He is a member of IET.

ELFATIH A. A. ELSHEIKH (Member, IEEE) received the B.Sc. degree in electrical and computer engineering from Omdurman Islamic University (OIU), Sudan, in 2000, the M.B.A. degree from the University of Khartoum (UoK), Sudan, in 2006, and the M.Sc. and Ph.D. degrees in electrical engineering from International Islamic University Malaysia (IIUM), in 2010 and 2017, respectively. He is currently an Assistant Professor with the Department of Electrical Engineering, College of Engineering, King Khalid University (KKU), Saudi Arabia. He has published more than 15 research papers in international journals and conferences. His research interests include wireless channel modeling, radio link design, RF propagation measurement, and modeling.

MD. RAFIQUL ISLAM (Senior Member, IEEE) received the B.Sc. degree in electrical and electronic engineering from BUET, Dhaka, in 1987, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Technology Malaysia, in 1996 and 2000, respectively. Currently, he is a Professor with the Department of Electrical and Computer Engineering, International Islamic University Malaysia. Over the course of his academic career, he has published more than 300 research papers in international journals and conferences. His primary research interests include antenna design, wireless channel modeling, RF and FSO propagation measurement, and modeling. He is a Life Fellow of the Institute of Engineers Bangladesh. He is a member of IET.

FAKHER ELDIN MOHAMED SULIMAN (Member, IEEE) received the B.Tech. degree in electrical engineering (control) from the Sudan University of Science and Technology, Khartoum, Sudan, in 1989, and the M.Eng. and Ph.D. degrees in electrical engineering (communication) from the University of Technology Malaysia, Johor Bahru, Malaysia, in 1989 and 2004, respectively. He is currently an Assistant Professor with the Electrical Engineering Department, College of Engineering, King Khalid University, Abha, Saudi Arabia. He conducts courses for both bachelor's and master's level students. He is the Coordinator of e-learning activities in his department. He is a quality matters certified master course reviewer for online courses. He has published a number of technical papers in international journals and conferences. His research interests include performance analysis and enhancement for wired and wireless communications systems, home automation systems, and optical switching networks. He is a Specialist Engineer of the Sudan Engineering Council and a full member of the Sudan Engineering Society. (Based on document published on December 2021).

SINIL MUBARAK received the Ph.D. degree in engineering from the Department of Electrical and Computer Engineering, Faculty of Engineering, International Islamic University Malaysia. His research interests include software engineering, computer network security, the Internet of Things, and artificial intelligence.