Cognitive Computation (2024) 16:2611–2625
https://doi.org/10.1007/s12559-024-10309-w

RESEARCH

Generative AI and Cognitive Computing-Driven Intrusion Detection System in Industrial CPS

Shareeful Islam[1] · Danish Javeed[2] · Muhammad Shahid Saeed[3] · Prabhat Kumar[4] · Alireza Jolfaei[5] · A.K.M. Najmul Islam[4]

Received: 28 December 2023 / Accepted: 13 May 2024 / Published online: 10 June 2024

Abstract
Industrial Cyber-Physical Systems (ICPSs) are becoming more and more networked and essential to modern infrastructure. This has led to an increase in the complexity of their dynamics and the challenges of protecting them from advanced cyber threats have escalated. Conventional intrusion detection systems (IDS) often struggle to interpret high-dimensional, sequential data efficiently and extract meaningful features. They are characterized by low accuracy and a high rate of false positives. In this article, we adopt the computational design science approach to design an IDS for ICPS, driven by Generative AI and cognitive computing. Initially, we designed a Long Short-Term Memory-based Sparse Variational Autoencoder (LSTM-SVAE) technique to extract relevant features from complex data patterns efficiently. Following this, a Bidirectional Recurrent Neural Network with Hierarchical Attention (BiRNN-HAID) is constructed. This stage focuses on proficiently identifying potential intrusions by processing data with enhanced focus and memory capabilities. Next, a Cognitive Enhancement for Contextual Intrusion Awareness (CE-CIA) is designed to refine the initial predictions by applying cognitive principles. This enhances the system's reliability by effectively balancing sensitivity and specificity, thereby reducing false positives. The final stage, Interpretive Assurance through Activation Insights in Detection Models (IAA-IDM), involves the visualizations of mean activations of LSTM and GRU layers for providing in-depth insights into the decision-making process for cybersecurity analysts. Our framework undergoes rigorous testing on two publicly accessible industrial datasets, ToN-IoT and Edge-IIoTset, demonstrating its superiority over both baseline methods and recent state-of-the-art approaches.

Keywords Computational design science · Cognitive computing · Generative AI · Intrusion detection system · Industrial cyber-physical systems

Introduction

The rapid technological advancement in communication, computing, data analytics, and storage enables industrial systems to migrate into digitalized, intelligent, and more reliable infrastructure. This led to the development of Industrial Cyber-Physical Systems (ICPSs) which couple physical with cyberinfrastructure by heavily relying on technology for reliable service delivery. Various industries across the sectors have now adopted ICPs specifically critical infrastructure sectors notably smart grid, transport, water management, and others [1, 2]. ICPSs are inherently complex with heterogeneous infrastructure and inter-connectivity among different cyber and physical sub-systems such as a generation plant of a smart grid is monitored and controlled by the SCADA system in real-time through substation communication standards and protocols, i.e., IEC 61850 and Modbus [3]. However, such inter-dependencies among various subsystems of the ICPSs increase the attack surface and introduce many security threats that can be exploited by the potential vulnerabilities and create cascading effects throughout the overall infrastructure which can degrade system efficiency and reliability, or even cause catastrophic consequences. There are numerous high-profile cyber attack examples that provide catastrophic impact on the ICPS and overall business notably the aluminum manufacturing company Norsk Hydro suffered with LockerGoga Ransomware attack in 2019 which severely impacted their entire global supply chain with 22,000 computers being hit across 170 different sites [4]. Further, a cryptocurrency-based malware attack on the SCADA of a European water utility company severely disrupted the distribution facility [5]. Therefore, the security of the ICPS is

Extended author information available on the last page of the article

paramount important for the resilience and survivability of the ICPS.

However, enhancing security for the ICPS is a challenging task for a number of reasons. Firstly, It is not always possible in real-time to update and restart industrial control devices due to their functionalities to monitor and control the physical part. For instance, SCADA is the heart of the smart grid transmission system which continuously monitors and regulates the distribution of electricity from the generation system using data from a Remote Terminal Unit (RTU) and Feeder Terminal Unit (FTU). Secondly, some of the devices need to comply with lower latency requirements which makes it challenging to incorporate additional security measures like encryption. Thirdly, there are unique threats and attack patterns for the sector-specific CPS such as vulnerabilities in smart grid infrastructure that are not similar to the transport system due to the distinct features of the cyber and physical infrastructure [6]. Finally, the threat landscape is continuously evolving with sophisticated attack patterns with numerous data to analyze that put significant challenges for managing the security of the CPS [7].

In this context, an intrusion detection system (IDS) is suggested as a fundamental step to secure the ICPSs by real-time monitoring of the traffic for detection of potential anomalies [8, 9]. However, traditional IDS used for the IT infrastructure need to be tailored for the ICPS due to the nature of the communication protocols, standards, and unique functionalities of the industrial devices. Several existing works focus on improving the detection techniques of IDS considering various physical and cyber sub-systems of ICPS such as anomaly-based network IDS which investigates the possible states of the various industrial control system [8], a report on reviewing of supervised-based intrusion detection system for SCADA emphasizing the necessity of processing power for detection intrusion [10], Smart Security Probe (S2P) is adopted to detect possible from the network and physical process of PLCs and SCADA systems [11], cognitive computing-based IDS [12, 13] and many more. Despite advancements in current intrusion detection approaches, several critical research gaps remain unaddressed, particularly in the realm of ICPSs [14]. Firstly, there is a notable deficiency in systems that can efficiently process and extract meaningful features from the high-dimensional, sequential data typical of ICPS environments, which is essential for identifying complex intrusion patterns [15]. Furthermore, a large number of present systems lack a substantial amount of contextual analysis, which raises the frequency of false positives and negatives. This deficiency highlights the need for approaches that can replicate human cognition and provide a more comprehensive contextual awareness of possible risks [16]. The interpretability of intrusion detection systems is another important gap. The absence of openness in the decision-making procedures of these systems frequently erodes confidence and makes it more difficult to develop sensible countermeasures [17]. Lastly, extensive testing and validation of IDS in various industrial settings are often neglected, raising doubts regarding their efficacy and dependability in practical situations. To improve IDS's effectiveness and agility in protecting the world's increasingly complex and integrated landscape, these shortcomings must be filled [18, 19].

To overcome the aforementioned shortcomings and improve intrusion detection in the ICPSs, this work employs the computational design science approach. Specifically, the proposed approach develops an intelligence IDS based on Generative AI and cognitive computing to facilitate a higher level of interpretability and transparency in the decision-making processes of the IDS.

Contribution

The main contributions of this article can be summarized as follows:

- **Advanced Feature Extraction with LSTM-SVAE**: Introduction of a Long Short-Term Memory-based Sparse Variational Autoencoder (LSTM-SVAE) for efficient feature extraction in ICPSs. This model leverages Generative AI to process high-dimensional, sequential data, providing a robust foundation for accurate anomaly detection.
- **Innovative Bidirectional RNN with Hierarchical Attention (BiRNN-HAID)**: Development of a novel Bidirectional Recurrent Neural Network enhanced with a hierarchical attention mechanism. This approach significantly improves the detection of complex intrusion patterns by focusing on pertinent features in the data.
- **Cognitive Enhancement for Contextual Intrusion Awareness (CE-CIA)**: Integration of cognitive computing elements for refining intrusion detection predictions. This stage adds a layer of context-aware analysis, reducing false positives and enhancing the overall reliability of the system.
- **Interpretive Assurance through Activation Insights (IAA-IDM)**: Implementation of a method to visualize and interpret activation patterns within the neural network. This transparency in the decision-making process enhances the interpretability of the IDS, providing cybersecurity analysts with valuable insights.

The remainder of this paper is organized as follows: The "Existing Literature" section presents the existing literature. In the "Research Design" section, we have discussed the research design used in this article. The performance evaluation is discussed in the "Performance Evaluation" section.

The "Conclusion" section concludes the paper with a future research perspective.

Existing Literature

Cybersecurity IS Literature and Computational Design Science Guidelines

In recent years, securing massive Information Systems (IS), such as extensive CPS and large-scale IIoT has been a gravitated research domain yielding multi-dimensional security approaches proposed to address diversified cybersecurity challenges [14]. In this context, DL-empowered security frameworks leveraged by their complex computational operations are significantly considered a prominent pathway to investigate adversarial elements in ICPSs [20]. Computational design science substantially provides valuable insights by embracing more logical and rational analysis of security problems associated with the ICPSs, leading toward more robust, appropriate, and trustworthy security solutions [21]. It further familiarizes with a set of methodologies and algorithms to enable the solution architects with a comprehensive understanding of the nature of knowledge to develop adequate solutions to human-centric problems sustainably [22, 23]. Literature has witnessed an abundance of research contributions to support this discussion. Researchers in [13] have designed an intelligent threat detection model under the norms and best practices of computational design science. The proposed model is remarkably strengthened by cognitive computing and aims to interrogate suspicious entities in ICPS. The framework is equipped with a chain of processes where the Binary Bacterial Frogging Optimization (BBFO) technique is adopted for effective feature extraction, Gated Recurrent Unit (GRU) is employed for classification, and Nesterov-Accelerated Adaptive Moment Estimation (NADAM) optimizer is applied to enhance the detection rate of GRU. The proposed system is trained on the CICIDS2017 and NSL-KDD datasets and is evaluated in terms of attack detection accuracy, precision, recall, and f1-score. Another cognitive computing-driven approach is applied in [24], where the authors have developed a novel detection mechanism to investigate perilous threat categories such as probe attacks, User-to-Root (U2R), Remote-to-Local (R2L) attacks, etc. The model is integrated with a Convolutional Neural Network (CNN) and is trained on an NSL-KDD dataset, carrying thousands of relevant threat impressions. Evaluation results validate the performance of the proposed model regarding the timely detection of attacking instances. CNN along with Graph Convolutional Network (GCN) is implied in another cognitive computing-based IDS designed to explore Advanced Persistent Threats (APT) [25]. The system continuously examines the functioning processes in endpoint systems to extract the malware behavior and aggregate it by employing GCN. After that, the CNN mechanism is applied to detect the APT malware by analyzing the malware instances collected by GCN. Experiments performed to evaluate the performance of the designed approach dignify its potential to detect APTs in endpoint systems.

Computational Models for Intrusion Detection

Cognitive computing-enabled approaches are attaining notable attention to develop reliable and consistent security solutions for broadly expanded ICPSs. The charismatic influence of cognitive computing is vividly reflected by its peculiar attributes, for example, adaptive learning, contextual understanding, human-centric predictions, scalable forensic capabilities, and automated response recommendations to countermeasure sophisticated threats. The authors in [26] proposed a DL-driven human cognitive privacy-preserving approach (DeepCog) with appropriate implementations in industrial policing. The designed framework is based upon a binary-facet Multi-layer Perceptron Neural Network (MLP-NN) that considers anonymized Electroencephalography (EEG) samples to ensure privacy by applying feature-transforming normalization. The PhysioNet BCI dataset contains Brain-Computer Interface (BCI)-derived EEG signal data. Researchers claim the significance of the proposed approach to enhancing trustworthiness in IIoT-enabled industrial policing. Deep Neural Network (DNN) is a notable DL classifier that has a variety of applications in designing security solutions for smart industries. Researchers in [27] have proposed a DNN-based on-demand communication system to analyze cognitive big data on edge devices in IIoT networks. The model is trained on the CIFAR100 dataset, which has data values from 100 classes, and is evaluated to get an idea of its efficiency in classifying big data. The authors suggest the implications of their model for security surveillance applications in large-scale IIoT communication scenarios. The authors in [28] present a DL-empowered model to investigate emerging cyber security attacks such as reconnaissance attacks, Complex Malicious Response Injection (CMRI) attacks, Naïve Malicious Response Injection (NMRI), Malicious State Command Injection (MSCI), etc, in scalable CPS. The model uses an LSTM classifier and is used on Industrial Control Systems (ICS). While inspecting the performance, the designed scheme has proven active resilience against the mentioned attack categories. Class imbalance is a crucial problem when designing intrusion detection solutions for CPS. Researchers in [29] have addressed this issue and introduced an Optimal Kernel Extreme Learning Machine (OKELM) in correspondence with an Imbalanced Generative Adversarial Network (IGAN) to efficiently investigate potential threats in real-time CPS environments. The designed scheme deploys an imbalanced data filter at the convolutional

layers and is trained on two datasets, e.g., the CICIDS2017 and KDDCup99 datasets. Experimental outcomes indicate the importance of the proposed scheme for efficiently detecting cyber threats. In addition to the class imbalance problem, False Data Injection (FDI) attacks are also considered an imperative attack category to disrupt the integrity of ICPS. Authors in [30] have addressed these issues by designing a generalized DNN-based attack detection approach aiming to identify varying sparsity of FDI attacks. The model is evaluated on IEEE power systems in various case studies where the experiments endorse its capacities for intrusion detection and handling large imbalances with high accuracy. However, the proposed scheme requires significant computational resources, declaring it an unfit choice for resource-constrained smart networks. Another attempt is made in [31], where researchers have employed CNN to formulate a K-fold Triplet CNN (KD-CNN) approach for the timely identification of suspicious elements in ICPS. The model aims to investigate several attack categories, including hulk, slowhttptest, slow loris, etc, under minimal consumption of system resources. The system is trained on CICIDS2017 and NSL-KDD datasets comprising numerous attacking impressions and their capability to address the crucial concerns of duplicate data and data redundancy. On the performance evaluation scale, the scheme proves an active protection shield for ICPS against vital cyber threats; however, significant communication latencies are also spotted.

Research Design

Proposed Cognitive Computing-Driven Interpretable Intrusion Detection in ICPSs

Stage 1: LSTM-Based Sparse Variational Autoencoder for Feature Extraction (LSTM-SVAE)

1. **LSTM Layer:** Long Short-Term Memory (LSTM) networks are a type of Recurrent Neural Network (RNN) designed for sequential data. Due to issues with the vanishing and exploding gradient, learning and remembering long-term dependencies in sequences is challenging for conventional RNNs. Thus, LSTM was developed to address these problems. The architecture of LSTM is made up of several cells, or repeating units. The input gate ($\mathbf{i_t}$), forget gate ($\mathbf{f_t}$) and output gate ($\mathbf{o_t}$) are each cell's three primary constituents. The LSTM can control the information flow by using these gates in conjunction with the cell state. The mathematical equations of an LSTM cell are as follows [32]:
    1. Forget Gate ($\mathbf{f_t}$): Decides how much of the previous cell state should be kept.

$$\mathbf{f_t} = \sigma(W^{\cdot}f[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_f}) \tag{1}$$

    2. Input Gate ($\mathbf{i_t}$): Decides what information about the new cell state to store.

$$\begin{aligned} \mathbf{i_t} &= \sigma(W^{\cdot}i[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_i}) \\ \hat{\mathbf{C}}t &= \tanh(W^{\cdot}C[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_C}) \end{aligned} \tag{2}$$

    3. Update Cell State: Integrate the $\mathbf{i_t}'s$ recommendation for the new cell state with the $\mathbf{f_t}'s$ decision.

$$\mathbf{C_t} = \mathbf{f_t} \odot \mathbf{C_{t-1}} + \mathbf{i_t} \odot \hat{\mathbf{C}}t \tag{3}$$

    4. Output Gate ($\mathbf{o_t}$): determines the following hidden state based on the input and the state of the cell.

$$\begin{aligned} \mathbf{o_t} &= \sigma(W^{\cdot}o[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_o}) \\ \mathbf{h_t} &= \mathbf{o_t} \odot \tanh(\mathbf{C_t}) \end{aligned} \tag{4}$$

    where $\mathbf{x_t}$ represents the current input, $\mathbf{h_{t-1}}$ is the hidden state of the previous cell, and $\mathbf{C_t}$ represents the cell state. Further, $W^{\cdot}f$, $W^{\cdot}i$, $W^{\cdot}C$, and $W^{\cdot}o$ are the weight matrices and $\mathbf{b_f}$, $\mathbf{b_i}$, $\mathbf{b_C}$, and $\mathbf{b_o}$ are its biases. Moreover, $\mathbf{h_t}$ denotes the hidden state and $\odot$ is the element-wise multiplication.
2. **SVAE Layer:** Given the LSTM's last $\mathbf{h_t}$ as an input, the encoder creates an output of the υ and σ^2 of $\mathbf{z}$.

$$\begin{aligned} \upsilon &= W_{\upsilon}\mathbf{h_t} + \mathbf{b}_{\upsilon} \\ \log\sigma^2 &= W_{\sigma}\mathbf{h_t} + \mathbf{b}_{\sigma} \end{aligned} \tag{5}$$

    where υ is the mean and σ^2 is the variance of $\mathbf{z}$: which is a latent variable. Further, W_{υ}, W_{σ} are the weights and $\mathbf{b}_{\upsilon}$, $\mathbf{b}_{\sigma}$ are the biases. Reparameterization trick is further used to sample the $\mathbf{z}$ as follow:

$$\mathbf{z} = \upsilon + \sigma \odot \epsilon \tag{6}$$

    Given the $\mathbf{z}$, the decoder then reconstructs the input sequence $\hat{x}$ as follow:

$$\hat{x} = f_d(\mathbf{z}) \tag{7}$$

    where f_d denotes the decoder function.
3. **Loss Function:** Furthermore, the loss function $\mathbf{L}$ is calculated using the following equation:

$$L = L_r + \beta L_{kl} + \lambda L_s \tag{8}$$

    where L_r denotes the reconstruction loss, L_{kl} represents the KL-divergence between the distribution of the encoder and a conventional normal distribution, and L_s is sparsity penalty. Further, the β and λ represent the hyperparameters. The working is explained in Algorithm 1.

Algorithm 1 LSTM-based sparse variational autoencoder for feature extraction (LSTM-SVAE)

LSTM layer for sequential data processing and feature extraction:
Initialize LSTM with input gate ($\mathbf{i_t}$), forget gate ($\mathbf{f_t}$), output gate ($\mathbf{o_t}$)
for each time step t **do**
  Compute forget gate: $\mathbf{f_t} = \sigma(W^{\cdot}f[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_f})$
  Compute input gate: $\mathbf{i_t} = \sigma(W^{\cdot}i[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_i})$
  Update candidate cell state: $\hat{\mathbf{C}}t = \tanh(W^{\cdot}C[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_C})$
  Update cell state: $\mathbf{C_t} = \mathbf{f_t} \odot \mathbf{C_{t-1}} + \mathbf{i_t} \odot \hat{\mathbf{C}}t$
  Compute output gate: $\mathbf{o_t} = \sigma(W^{\cdot}o[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_o})$
  Update hidden state: $\mathbf{h_t} = \mathbf{o_t} \odot \tanh(\mathbf{C_t})$
end for
SVAE Layer for Latent Variable Representation:
Given last hidden state $\mathbf{h_t}$ from LSTM:
Compute mean υ and variance $\log\sigma^2$ of latent variable $\mathbf{z}$
Sample $\mathbf{z}$ using reparameterization trick: $\mathbf{z} = \upsilon + \sigma \odot \epsilon$
Reconstruct input sequence $\hat{x}$: $\hat{x} = f_d(\mathbf{z})$
Calculation of Loss Function:
Calculate loss L: $L = L_r + \beta L_{kl} + \lambda L_s$

Stage 2: Bidirectional RNN with Hierarchical Attention for Intrusion Detection (BiRNN-HAID)

We have further employed Bidirectional LSTM and bidirectional GRU with a Hierarchical Attention mechanism for efficient intrusion detection. The details are as follows:

1. **Bidirectional LSTM:** A bidirectional LSTM has two parts,i.e., forward LSTM and backward LSTM. The input sequence is processed in opposing directions by each component. Equations (1) to (4) are the mathematical operations of the cell of an LSTM. For BiLSTM, it uses the following equation:

$$\begin{aligned}\overrightarrow{\mathbf{h_t}}^{frwd}, \overrightarrow{\mathbf{C_t}}^{frwd} &= LSTM(\overrightarrow{\mathbf{x_t}}, \overrightarrow{\mathbf{H_{t-1}}}^{frwd}, \overrightarrow{\mathbf{C_{t-1}}}^{frwd}) \\ \overleftarrow{\mathbf{h_t}}^{bkwd}, \overleftarrow{\mathbf{C_t}}^{bkwd} &= LSTM(\overleftarrow{\mathbf{x_t}}, \overleftarrow{\mathbf{H_{t+1}}}^{bkwd}, \overleftarrow{\mathbf{C_{t+1}}}^{bkwd})\end{aligned} \tag{9}$$

   where $\overrightarrow{\mathbf{h_t}}^{frwd}$ represents the hidden state and $\overrightarrow{\mathbf{C_t}}^{frwd}$ denotes cell state of the forward LSTM at time step t. However, $\overleftarrow{\mathbf{h_t}}^{bkwd}$ represents the hidden state and $\overleftarrow{\mathbf{C_t}}^{bkwd}$ denotes cell state of the backward LSTM at time step t. Further, $\overrightarrow{\mathbf{x_t}}$ and $\overleftarrow{\mathbf{x_t}}$ denotes the input for forward and backward LSTM at t respectively.
2. **Bidirectional GRU:** A bidirectional GRU also has two parts: forward and backward GRU. The forward GRU processes the input in the forward while the backward processes it in the backward direction. A simple GRU comprises two gates, i.e., update ($\mathbf{z_t}$) and reset gate ($\mathbf{r_t}$) with a candidate state ($\hat{\mathbf{h}}_t$) and updated hidden state ($\mathbf{h_t}$). The following are the mathematical operations of a GRU cell [33]:

$$\begin{aligned}\mathbf{z_t} &= \sigma(W^{\cdot}z[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_z}) \\ \mathbf{r_t} &= \sigma(W^{\cdot}r[\mathbf{h_{t-1}}, \mathbf{x_t}] + \mathbf{b_r}) \\ \hat{\mathbf{h}}t &= \tanh(W^{\cdot}h[\mathbf{r_t} \odot \mathbf{ht - 1}, \mathbf{x_t}] + \mathbf{b_h}) \\ \mathbf{h_t} &= (1 - \mathbf{z_t}) \odot \mathbf{h_{t-1}} + \mathbf{z_t} \odot \hat{\mathbf{h}}_t\end{aligned} \tag{10}$$

   where $\mathbf{x_t}$ denotes the input, $\mathbf{h_t}$ is the hidden state, σ represents the sigmoid activation function. $W^{\cdot}z$, $W^{\cdot}r$, and $W^{\cdot}h$ are the weight matrices and b_z, b_r, and b_h are bias vectors. The bidirectional GRU computes the operations of forward and backward GRU by using the equation as follows:

$$\begin{aligned}\overrightarrow{\mathbf{h_t}}^{frwd} &= GRU(\overrightarrow{\mathbf{x_t}}, \overrightarrow{\mathbf{h_{t-1}}}^{frwd}) \\ \overleftarrow{\mathbf{h_t}}^{bkwd} &= GRU(\overleftarrow{\mathbf{x_t}}, \overleftarrow{\mathbf{h_{t+1}}}^{bkwd})\end{aligned} \tag{11}$$

   where $\overrightarrow{\mathbf{h_t}}^{frwd}$ and $\overleftarrow{\mathbf{h_t}}^{bkwd}$ represents the forward and backward GRU hidden states, while $\overleftarrow{\mathbf{x_t}}$ and $\overleftarrow{\mathbf{x_t}}$ are the inputs.
3. **Attention Mechanism:** Further, the method used to calculate a sequence's attention scores is:

$$\sigma_t = SMax(W^{\cdot}a.\mathbf{h_{t-1}} + \mathbf{b_a}) \tag{12}$$

   The context vector of the sequence becomes:

$$\mathbf{c} = \sum_t \sigma_t \mathbf{h_t} \tag{13}$$

4) Dense Output Layer: Moreover, we employed a dense layer, where the flattened features from the attention go through a linear transformation and then an activation:

$$M = \sigma(W^{\cdot}d.\mathbf{f} + \mathbf{b_d}) \tag{14}$$

where σ represents the softmax activation function for multiclass classification. The following equation performs the operation for softmax:

$$\mathbf{p_i} = \frac{\mathbf{e}^z i}{\sum_{\mathbf{n}}^{\mathbf{j=1}} \mathbf{e}^z j} \tag{15}$$

where $\mathbf{p_i}$ denotes the probability of the input belongs to class i, zi is the logit and $\mathbf{n}$ represents the total number of classes. The working is explained in Algorithm 2.

Stage 3: Cognitive Enhancement for Contextual Intrusion Awareness (CE-CIA)

The "Generalized Cognitive Refinement of Confidence Scores" Algorithm 3, positioned as Stage 3 in the CE-CIA framework, is a significant advancement in the realm of IDS.

Algorithm 2 Bidirectional LSTM and GRU with hierarchical attention mechanism

Bidirectional LSTM:
for each time step t in input sequence **do**
  Compute forward LSTM: $\overrightarrow{\mathbf{h_t}}^{frwd}, \overrightarrow{\mathbf{C_t}}^{frwd}$
  Compute backward LSTM: $\overleftarrow{\mathbf{h_t}}^{bkwd}, \overleftarrow{\mathbf{C_t}}^{bkwd}$
end for
Bidirectional GRU:
for each time step t in input sequence **do**
  Compute forward GRU: $\overrightarrow{\mathbf{h_t}}^{frwd}$
  Compute backward GRU: $\overleftarrow{\mathbf{h_t}}^{bkwd}$
end for
Attention Mechanism:
for each time step t **do**
  Compute attention scores: $\sigma_t = SMax(W_a \mathbf{h_{t-1}} + \mathbf{b_a})$
  Calculate context vector: $\mathbf{c} = \sum_t \sigma_t \mathbf{h_t}$
end for
Dense Output Layer:
Flatten features and apply linear transformation
Apply softmax activation: $\mathbf{p_i} = \frac{\mathbf{e}^{z}i}{\sum_{\mathbf{j=1}}^{\mathbf{n}} \mathbf{e}^{z}j}$

This stage is instrumental in fine-tuning the confidence scores derived from predictive models, thereby elevating the overall accuracy and reliability of intrusion detection. intrusion detection systems are often challenged by the delicate balance between accurately identifying genuine threats (true positives) and avoiding false alarms (false positives). The CE-CIA stage addresses this challenge by applying a cognitive layer of analysis to the preliminary results obtained from earlier stages of the IDS. This layer is not just a filter but a cognitive enhancer that intelligently refines the confidence scores based on contextual understanding. The working of the algorithm is both methodical and insightful. It begins by iterating over the set of initial prediction results. For each instance, the algorithm assesses whether the associated confidence score falls below a predefined threshold. If it does, this is interpreted as an indication of uncertainty, and the score is conservatively reduced by a specified factor. This reduction is rooted in the cognitive principle of minimizing false positives, particularly in ambiguous cases where the model's confidence is not high enough. Conversely, for instances where the model exhibits high confidence, the scores are maintained, signifying a clear pattern recognition by the model. The system creates an updated set of cognitively improved predictions by adding these refined scores to the findings. This shows that they are assessments that take into account the finer points and circumstances of possible intrusion scenarios rather than just being numerical values. Due to this, the output provides an improved understanding of the probability of an intrusion, enabling cybersecurity experts to make more informed decisions.

Algorithm 3 Generalized cognitive refinement of confidence scores

procedure CognitiveRefineConfidence($results$, $threshold$, $factor$)
  $refinedResults \leftarrow$ empty list ▷ Initial parameters: ▷ $results$ - DataFrame of initial prediction results ▷ $threshold$ - Confidence score threshold for refinement ▷ $factor$ - Reduction factor for scores below the threshold ▷ Iterate over results to apply context-aware refinement
  for each row in $results$ **do** ▷ Evaluate if refinement is needed based on cognitive criteria
    if $row.ConfidenceScore < threshold$ **then**
      $refinedScore \leftarrow row.ConfidenceScore \times factor$ ▷ Reduce score to minimize false positives, reflecting contextual understanding
    else
      $refinedScore \leftarrow row.ConfidenceScore$ ▷ Maintain high confidence scores, indicative of clear patterns
    end if ▷ Accumulate refined results with cognitive insights
    Append $(row.ID, row.TrueLabel, refinedScore)$ to $refinedResults$
  end for
  return $refinedResults$ as DataFrame ▷ Output provides a cognitive-enhanced view of intrusion likelihood
end procedure

Stage 4: Interpretive Analysis and Assurance of Intrusion Detection Mechanisms (IAA-IDM)

Intrusion detection, a critical component in cybersecurity, involves analyzing network data to identify potential unauthorized or malicious activities. The complexity and evolving nature of network intrusions necessitate models that not only detect but also provide insights into their decision-making processes. The Algorithm 4 plays a pivotal role in enhancing the effectiveness of IDS by interpreting activation values in Bi-directional Long Short-Term Memory (BiLSTM) and Bi-directional Gated Recurrent Unit (BiGRU) layers. This algorithm is designed to meet a specific need. It starts by

Algorithm 4 Interpretation of activation values for BiRNN-HAID

Input: Trained neural network model with BiLSTM and BiGRU layers
Output: Mean activation values for BiLSTM and BiGRU units
Select a subset of the data for activation visualization
Define $subsetSize$ to specify the number of samples
Extract a subset of the input data, $xSubset$, of size $subsetSize$
Create an activation model to extract intermediate activations
Define $activationModel$ to output activations from BiLSTM and BiGRU layers
Get activations for the subset
$activations \leftarrow activationModel.predict(xSubset)$
Calculate mean activation values for BiLSTM and BiGRU units
$meanActivationsBiLSTM \leftarrow activations[0].mean(axis = 0)$
$meanActivationsBiGRU \leftarrow activations[2].mean(axis = 0)$
Visualize the mean activation values
Create bar plots for $meanActivationsBiLSTM$ and $meanActivationsBiGRU$
Label the x-axis as "Unit" and y-axis as "Mean Activation Value"

selecting a subset of network data, focusing on instances that are representative of typical network traffic. The core of the algorithm involves a specialized neural network model, composed of BiLSTM and BiGRU layers, designed to process and analyze this data. 'BiLSTMs and BiGRUs are adept at handling sequential data, making them ideal for analyzing time-dependent network traffic patterns. Their unique architecture allows them to remember long-term dependencies and nuances in the data, crucial for detecting sophisticated intrusion patterns. This approach is novel since it can interpret these BiLSTM and BiGRU layer activations. The approach gives us a better understanding of the model's focus during prediction by determining the mean activation values of each unit within these layers. These activation values provide a window into the "thought process" of the model by essentially indicating the contribution of each unit to the choice made at a certain time step. The algorithm's usefulness is further increased through the visualization of these mean activation levels. Cybersecurity analysts can determine which characteristics or patterns in network traffic are most important for identifying breaches by charting these values. This knowledge is extremely helpful for comprehending the behavior of the model, optimizing its functionality, and even directing the creation of more potent intrusion detection techniques.

Performance Evaluation

This section provides the complete details about the experimental setup followed by the dataset and pre-processing details. We further provide details about the metrics that are employed to evaluate the proposed model's performance. Finally, we evaluate the performance of the proposed IDS and discuss the results in this section.

Experimental Setup

The experiment is conducted on a PowerEdge R940xa Rack Server, equipped with two Intel Xeon Gold 6240 processors running at 2.6 GHz, 256 GB of RAM, and 8 NVIDIA Ampere A100, 80GB Passive GPUs. The server uses Windows Server 2019 standard. The deep learning models are built using TensorFlow 2.16 and Keras 3. In order to select the most suitable parameters, we conducted numerous experiments (approximately 5 to 7 iterations) guided by the results of performance metrics. The final parameters used are illustrated in Table 1. Additionally, the default parameters of Decision Tree (DT), Random Forest (RF), and Naive Bayes (NB) in scikit-learn Python are utilized.

Table 1 Experimental setup for each stage

Stage 1: LSTM-SVAE	
LSTM component	
LSTM layers	3
Units per LSTM layer	128
Output layer function	Softmax
Input layer nodes	44 for ToN-IoT and 96 for Edge-IIoTset
LSTM timestep	1
SVAE component	
Encoder layers	2 (128 and 32 nodes)
Encoder activation	tanh
Decoder layers	2 (32 and 128 nodes)
Decoder activation	tanh
Loss	Kullback-Leibler Divergence
Epochs	2000
Optimizer	Adam
Batch size	250
Stage 2: BiRNN-HAID	
BiRNN layers	2
Number of units in each BiRNN layer	64
Attention mechanism	Hierarchical
Attention type	self-attention
Output layer function	Softmax
Input layer nodes	10 for both datasets
BiRNN timestep	1
Optimizer	Adam
Loss	Categorical cross-entropy

Dataset and Preprocessing

We employed two publicly available datasets, such that ToN-IoT [34] and Edge-IIoTset [35] to evaluate the performance of the proposed IDS. ToN-IoT is a significant resource for research in the field of IT security. It is designed to facilitate the study of IDS by providing a comprehensive set of network traffic data that simulates a variety of cyber-attacks and normal traffic scenarios. This dataset is instrumental in developing and evaluating IDS models. On the other hand, the EDGEIIoTset dataset is particularly focused on edge computing environments within the CPS ecosystem. It provides data related to CPS devices operating in edge computing scenarios, including network traffic, device behavior, and security threats specific to such environments. Further, it aids in the development of security solutions and monitoring systems that are optimized for the edge computing landscape.

In this work, we consider a normal class and nine attack classes of the ToN-IoT dataset, i.e., DDoS, Backdoor, MiTM, etc while for the Edge-IIoTset we consider one normal and fourteen attack classes. Furthermore, we employed different steps to preprocess the data as it impacts the performance of the model [36]. Firstly, we imputed all the missing values and removed the incomplete rows from both the dataset. Secondly, we converted all the categorical variables to numerical values by using one-hot encoding. Thirdly, we employed the Min-Max scaler function to normalize the data. Finally, we divide both the datasets into training and testing data, i.e., the model was trained using 70% of the data and validated and tested using the remaining 30%. The complete details about the instances in the training and testing sets of these datasets are provided in Table 2.

Evaluation Metrics

In this study, we used a number of assessment measures, including Accuracy (Acc), Precision (Pr), Recall (Re), and F1-score (F1), to evaluate the performance of the proposed IDS. For additional performance assessment, we used the confusion matrix and Receiver Operating Characteristic (ROC) curve. The following equations are used to calculate the values for Acc, Pr, Re, and F1 [37]:

$$Acc = \frac{T_r P + T_r N}{T_r P + T_r N + F_a P + F_a N} \tag{16}$$

$$Pr = \frac{T_r P}{T_r P + F_a P} \tag{17}$$

$$Re = \frac{T_r P}{T_r P + F_a N} \tag{18}$$

$$F1 = 2 \times \frac{Pr \times Re}{Pr + Re} \tag{19}$$

where $T_r P$ denotes the true positive, $T_r N$ represents the true negative, $F_a P$ is the false positive, and $F_a N$ is false negative. Further, for overall analysis, we have used weighted component. The weighted calculations for precision, recall, and F1-score are as follows: The precision for each class is weighted by the number of true instances for that class in the dataset. The overall weighted precision is the sum of these individual weighted precisions.

$$Pr_{\text{weighted}} = \sum_{i=1}^{N} w_i \times Pr_i \tag{20}$$

where w_i is the proportion of true instances for class i in the dataset, and Pr_i is the precision for class i. N is the total number of classes. The recall for each class is weighted by the proportion of true instances for that class. The overall weighted recall is the sum of these individual weighted recalls.

$$Re_{\text{weighted}} = \sum_{i=1}^{N} w_i \times Re_i \tag{21}$$

where w_i is as defined above, and Re_i is the recall for class i. The F1-score for each class is computed and then weighted by the proportion of true instances for that class. The overall weighted F1-score is the sum of these individual weighted F1-scores.

$$F1_{\text{weighted}} = \sum_{i=1}^{N} w_i \times F1_i \tag{22}$$

Table 2 Datasets detail

Dataset	Class	Training set	Testing set
ToN-IoT	Backdoor	12,991	6009
	DDoS	13,984	6016
	DoS	13,951	6049
	Injection	13,952	6048
	MiTM	733	310
	Normal	209,964	90,036
	Password	14,100	5900
	Ransomeware	14,011	5989
	Scanning	14,032	5968
	XSS	14,012	5988
Edge-IIoTset	Normal	966,595	414,263
	DDoS UDP	85,319	36,248
	DDoS ICMP	47,626	20,313
	SQL Injection	35,626	15,200
	DDoS TCP	34984	15,078
	Vulnerability scanner	34,735	15,291
	Password	34,833	15,100
	DDoS HTTP	34,508	14,695
	Uploading	25,835	11,080
	Backdoor	16,830	7196
	Port scanning	14,082	5901
	XSS	10,510	4556
	Ransomware	6803	2886
	Fingerprinting	574	279
	MITM	252	106

Performance Evaluation of the proposed IDS

We evaluate the performance of the proposed IDS in this subsection. Firstly, we provide the accuracy vs loss output of the proposed model to show the optimal fit. Figure 1 depicts the

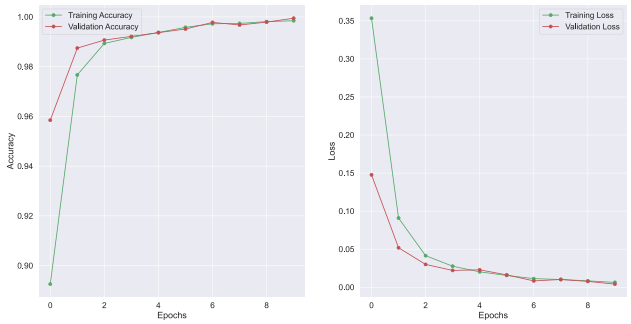

Fig. 1 Acc vs Loss ToN-IoT

training Acc and validation Acc Vs training loss and validation loss for the ToN-IoT dataset. In contrast, Fig. 2 presents the output for the Edge-IIoTset dataset. It can be seen in Fig. 1 that the proposed model achieved a training and validation Acc of 99.85% and 99.95% for the ToN-IoT dataset, while it has a training loss of 0.64% with validation loss of 0.41% respectively. For the Edge-IIoTset dataset, it achieved training Acc of 95.32% and validation Acc of 95.35% with training and validation loss of 9.35% and 9.30% accordingly. These results show the optimal fit of the proposed model and prove it is neither overfitting nor underfitting. Further, a confusion matrix, which is also known as an uncertainty matrix is used for evaluation. In the confusion matrix, each of the rows denotes the true class and the predicted class is represented by each column in the matrix. The cell indicates the number of instances from the true class that were predicted correctly by the model. We provide the confusion matrix of the proposed mechanism using both datasets. Figure 3 depicts the confusion matrix for the Ton-IoT dataset, and Fig. 4 presents the confusion matrix for the Edge-IIoTset dataset. It can be seen that the proposed model identified all of the classes of these datasets correctly, i.e., it predicted 90,036 instances from the Normal class, 6031 from DoS, 6014 from the DDoS class, and so on. Moreover, the Receiver Operating Characteristic (ROC) is also considered an important evaluation metric. It is a graphical representation, which is used to evaluate the performance of a classification model. An ROC value near

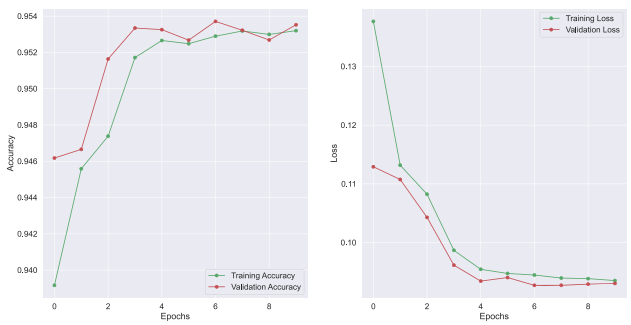

Fig. 2 Acc vs Loss Edge-IIoT

Fig. 3 This confusion matrix provides an in-depth evaluation of the model's classification performance for various classes present in ToN-IoT dataset. The *x*-axis represents predicted labels, while the *y*-axis corresponds to true labels

1 indicates the efficient performance of a model, while ROC values less than 0.5 are considered as poor performance by the model. We provide the ROC curve of the proposed model in Figs. 5 and 6 for ToN-IoT and Edge-IIoTset datasets. It can be seen that the proposed model has a 0.99999 micro average and a 0.99998 macro average for the ToN-IoT dataset. Further, it has micro and macro averages of 0.99931 and 0.99522 for the Edge-IIoTset dataset respectively. The micro and macro averages under both these datasets are almost equal to 1, which further indicates the efficient performance of the proposed IDS.

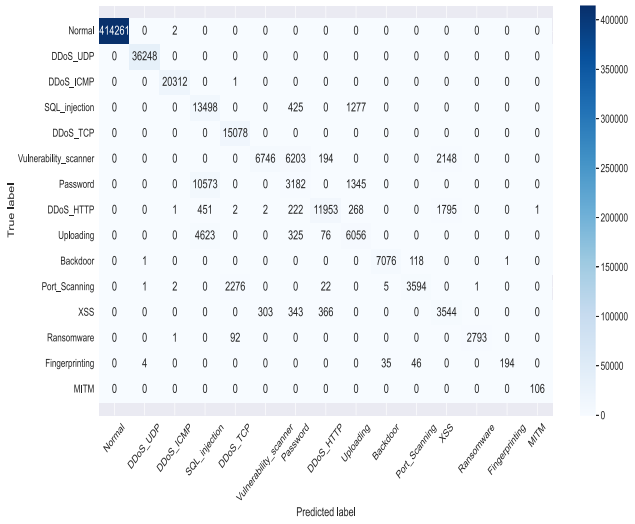

Fig. 4 This confusion matrix provides an in-depth evaluation of the model's classification performance for various classes present in Edge-IIoT dataset. The *x*-axis represents predicted labels, while the *y*-axis corresponds to true labels

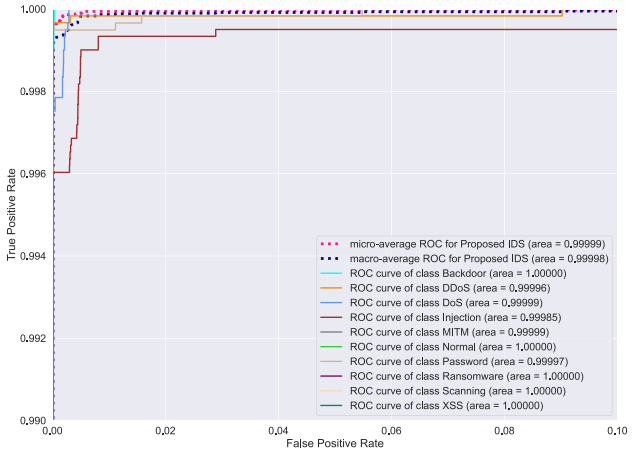

Fig. 5 In this ROC curve, each class from ToN-IoT dataset is evaluated based on its False Positive Rate (FPR) depicted on the *x*-axis and True Positive Rate (TPR) represented on the *y*-axis

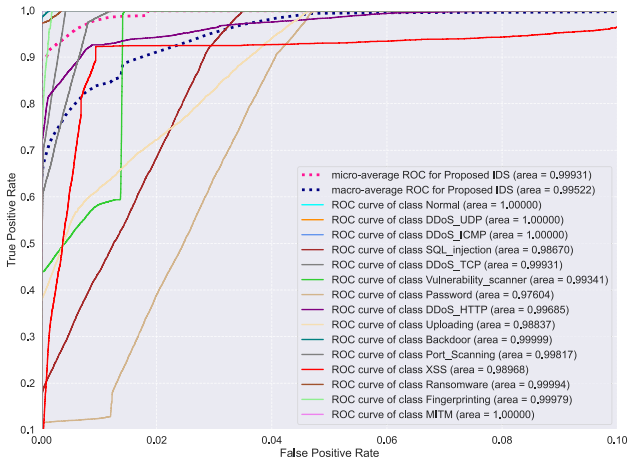

Fig. 6 In this ROC curve, each class from Edge-IIoT dataset is evaluated based on its False Positive Rate (FPR) depicted on the *x*-axis and True Positive Rate (TPR) represented on the *y*-axis

Moreover, we provide the class-wise performance of the proposed IDS in terms of Pr, Re, and F1. Table 3 presents the class-wise performance of the proposed IDS using the ToN-IoT dataset. The proposed IDS has a Pr of 100% for the Backdoor, Normal, and Scanning classes. For other classes, it has achieved Pr values between 98.07 and 99.96%. In terms of Re, it has achieved 100% Re for Normal, Ransomware, Scanning, and XSS classes. However, it has Re between 99.58 and 99.96% for the remaining classes of the ToN-IoT dataset. For F1, it has achieved F1 of 99.88% for Backdoor class, 99.78% for DDoS, 99.83% for DoS, 99.64% for Injection, 98.39% for MiTM, 99.90% for Password, 99.89% for Ransomware, and 99.94% for XSS classes. It has achieved an F1 of 100% for Normal and Scanning classes. Regarding the false positive rate, the proposed IDS achieved the lowest false positive rate of 0.00 for the MITM class. For other classes, it has a false positive rate between 0.000001 and 0.00005. Furthermore, we provide the class-wise performance for the Edge-IIoTset dataset in Table 4. Regarding Pr, it achieved the Pr of 100% for the Normal class, while for DDoS UDP, it has Pr of 99.98%. Further, it has a Pr of 99.97% for DDoS ICMP, 46.31% for SQL Injection, 86.41% for DDoS TCP, 95.67 for Vulnerability Scanner, 29.73% for Password, 94.78% for DDoS HTTP, 67.69% for Uploading, 99.43 for Backdoor, 95.63% for Port Scanning, 47.33% for XSS, 99.96% for Ransomware, 99.48% for Fingerprinting, and 99.06% for MITM classes. Regarding Re, it has 100% Re for DDoS UDP, DDoS TCP, and MITM classes. For other classes, it has a minimum Re of 21.07% for the Password class and a maximum of 97.77% for the Ransomware class. Moreover, it has achieved an F1 of 99.99% for the Normal and DDoS UDP classes, 99.98% for DDoS ICMP, 60.87% for SQL Injection, 92.71% for DDoS TCP, 60.38 for Vulnerability Scanner, 24.66% for Password, 87.54% for DDoS HTTP, 60.48% for Uploading, 98.88 for Backdoor, 74.41% for Port Scanning, 58.85% for XSS, 98.34% for Ransomware, 81.85$ for Fingerprinting, and 99.53% for MITM classes. Moreover, the proposed IDS achieved the lowest false positive rate of 0.00 for the Normal and MITM classes. For other classes, it has a false positive rate between 0.000001 and 0.03186.

Analysis for Generalized Cognitive Refinement of Confidence Score

Figures 7 and 8 compare the initial confidence scores (Stage 2) with the refined scores post-algorithm application (Stage 3) for "password" and "DDoS_UDP" attack present in ToN-IoT and Edge-IIoTset datasets. This approach can be generalized for each attack present in the dataset. The notable differences between these two stages, particularly the reduction in confidence scores for several instances, indi-

Table 3 Class-wise results (%) for ToN-IoT dataset

Parameters	Backdoor	DDoS	DoS	Injection	MITM	Normal	Password	Ransomware	Scanning	XSS
Precision (Pr)	100.00	99.60	99.96	99.70	98.07	100.00	99.94	99.80	100.00	99.89
Recall (Re)	99.76	99.96	99.70	99.58	98.70	100.00	99.86	100.00	100.00	100.00
F1-score (F1)	99.88	99.78	99.83	99.64	98.39	100.00	99.90	99.89	100.00	99.94
False positive rate	0.00005	0.00013	0.00002	0.00012	0.00	0.00004	0.00002	0.00014	0.00001	0.000007

Table 4 Class-wise results (%) for Edge-IIoTset dataset

Parameters	Normal	DDoS UDP	DDoS ICMP	SQL Injection	DDoS TCP	Vulnerability Scanner	Password	DDoS HTTP	Uploading	Backdoor	Port Scanning	XSS	Ransomware	Fingerprinting	MITM
Precision (Pr)	100.00	99.98	99.97	46.31	86.41	95.67	29.73	94.78	67.69	99.43	95.63	47.33	99.96	99.48	99.06
Recall (Re)	99.99	100.00	99.99	88.80	100.00	44.11	21.07	81.34	54.65	98.33	60.90	77.78	96.77	69.53	100.00
F1-score (F1)	99.99	99.99	99.98	60.87	92.71	60.38	24.66	87.54	60.48	98.88	74.41	58.85	98.34	81.85	99.53
False positive rate	0.00	0.000001	0.000005	0.00046	0.00452	0.00076	0.03186	0.00092	0.00248	0.00009	0.00029	0.00671	0.000003	0.000003	0.00

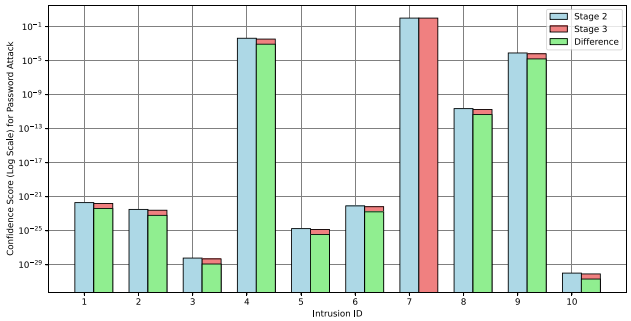

Fig. 7 Cognitive refinement of confidence score for password attack present in ToN-IoT dataset

cate the algorithm's conservative approach toward instances with lower initial confidence. This approach is especially evident where the initial confidence scores are significantly reduced post-refinement, aligning with the algorithm's criterion of scaling down scores below the threshold. This outcome demonstrates the efficacy of the cognitive refinement process in enhancing the reliability of the intrusion detection system. By applying this algorithm, we ensure that the system's predictions are not just based on initial assessments but are re-evaluated through a lens that mimics human-like skepticism and caution. Consequently, this method aids in reducing false positives, thereby strengthening the system's capability to differentiate between genuine threats and benign activities.

Interpretation of Activation Values

In Figs. 9 and 10, the *x*-axis represents the individual units of the LSTM layer. Since the LSTM layer was defined with 64 units and is bidirectional, it effectively has 128 units (64 in each direction). The *y*-axis, values represent the mean activation of each LSTM unit over the subset of data processed. Activation values in LSTM units can be negative or positive, indicating the extent to which each unit is activated by the input data. Values close to 0 suggest minimal activation, while higher absolute values (either positive or negative)

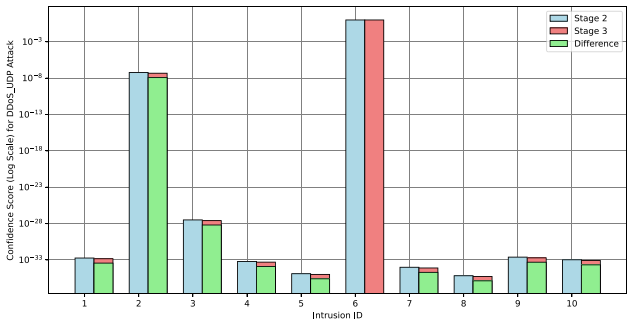

Fig. 8 Cognitive refinement of confidence score for DDoS_UDP attack present in Edge-IIoTset dataset

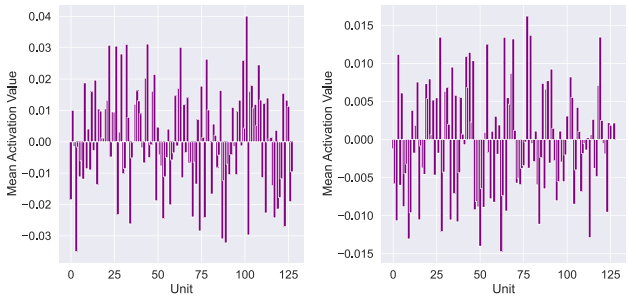

Fig. 9 Mean activation values ToN-IoT dataset

indicate stronger activation. By comparing two parts inside Figs. 9 and 10 for ToN-IoT and Edge-IIoT datasets, we can get insights into how different types of RNN units (LSTM vs. GRU) process the same data. It might reveal differences in how they capture and respond to patterns in the data. Understanding the activation patterns can also help in diagnosing the model's behavior. For example, if certain units are consistently not activated (mean activation values close to 0), they might not be contributing much to the model's performance. Moreover, high variation in activation values across the units may also indicate how different units are picking up various features or aspects of the input data. This can lead to insights into which features are more relevant to the model's predictions.

Comparison with Baselines

Finally, the performance of the proposed IDS is compared with some baseline approaches, i.e., Decision Tree (DT), Random Forest (RF), Naive Bayes (NB), Long-short-term Memory (LSTM), and Bidirectional Long-short-term Memory (BiLSTM) to further evaluate its performance. The comparison with these baseline approaches on the ToN-IoT dataset is provided in Fig. 11. It is clear from the figure that the proposed IDS obtained an Acc of 99.95% with Pr, Re, and F1 each at 99.94%. On the other hand, DT has an Acc, Pr, Re, and F1 of 95.34%, 74.72%, 80.00%, and 76.33%. Further, the RF has Acc of 97.81%, NB has 90.69%, LSTM has 82% and BiLSTM has 84.49%. Whereas the Pr values of RF, NB, LSTM,

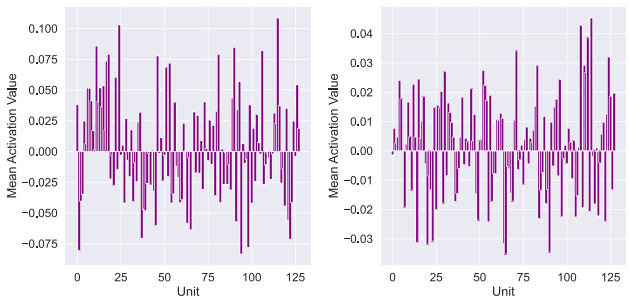

Fig. 10 Mean activation values Edge-IIoT dataset

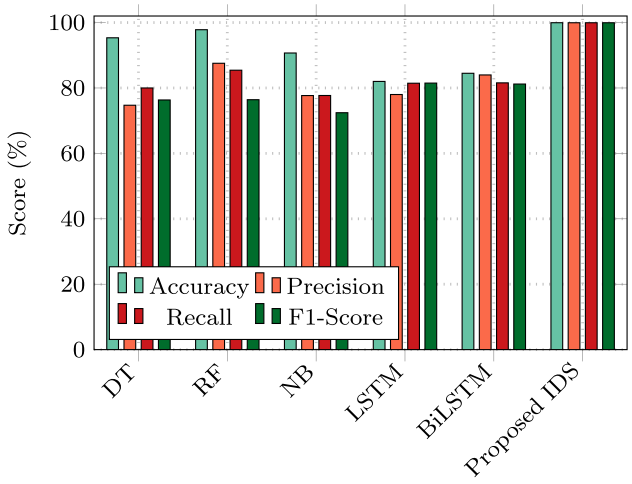

Fig. 11 Comparison of algorithm performance on ToN-IoT dataset

and BiLSTM are 87.55%, 77.68%, 78.00%, and 83.98% accordingly. Regarding Re, they achieved Re of 85.43%, 77.70%, 81.45%, and 81.56%. Finally, they have F1 values of 76.41%, 72.43%, 81.49%, and 81.20%. The proposed IDS outperformed the baseline classifiers by achieving higher values of Acc, Pr, Re, and F1 under the ToN-IoT dataset.

We further provide a comparison against these baseline approaches using the Edge-IIoTset dataset. Figure 12 depicts the comparison of the proposed IDS against these baselines. It can be seen that the proposed IDS has an Acc of 94.20%, Pr of 95.06%, Re of 94.19%, and F1 of 94.07%. The values of Acc achieved by the baselines are as follows: DT achieved 92.20%, RF achieved 92.50%, NB achieved 92.00%, LSTM achieved 92.80%, and BiLSTM achieved an Acc of 93.00% accordingly. Regarding Pr, the DT has Pr of 93.06%, whereas the Pr values of RF, NB, LSTM, and BiLSTM are 93.36%, 92.86%, 93.6%, and 93.86%. Furthermore, the Re values of

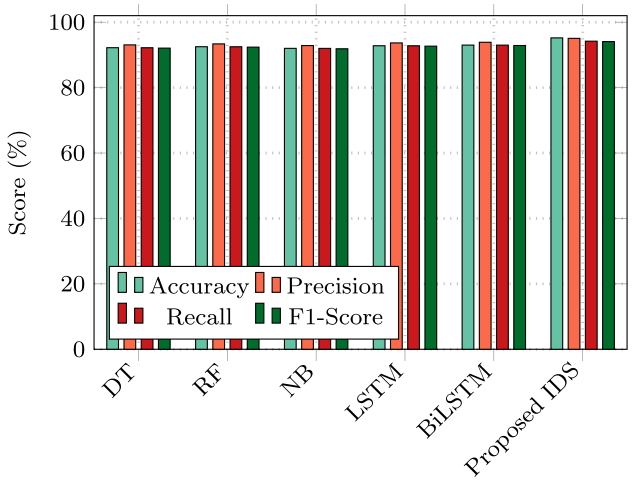

Fig. 12 Comparison of algorithm performance on Edge-IIoT dataset

these baseline approaches are as follows: DT has achieved Re of 92.19%, RF has 92.49%, NB has 91.99%, LSTM has 92.79%, and BiLSTM has Re of 92.99% respectively. Finally, the comparison in terms of F1 under the Edge-IIoTset dataset is also provided in Fig. 12. The DT and RF have F1 values of 92.07% and 92.37%. While, NB, LSTM, and BiLSTM have F1 of 91.87%, 92.67%, and 92.87% accordingly. This comparison using the Edge-IIoTset dataset also proves the efficient performance of the proposed IDS compared to these baseline approaches, thus proving its efficiency in threat detection.

Conclusion

In order to improve intrusion detection in Industrial Cyber-Physical Systems (ICPSs), this research introduces a unique approach that uses Generative AI and cognitive computing. For effective data processing and feature extraction, the system uses a Long Short-Term Memory-based Sparse Variational Autoencoder (LSTM-SVAE), and for precise detection of complex intrusion patterns, it uses a Bidirectional RNN with Hierarchical Attention (BiRNN-HAID). The Cognitive Enhancement for Contextual Intrusion Awareness (CECIA) component improves threat understanding and reduces false positives and negatives. The Interpretive Assurance through Activation Insights in Detection Models (IAA-IDM) provides insights into the system's decision-making process, enhancing its transparency and trustworthiness. Future efforts will focus on refining the proposed method to strike an ideal balance between detection accuracy and computational efficiency. This will include using machine learning to allocate resources more intelligently, improving algorithms for efficiency without compromising quality, and testing different configurations to identify the most effective approach. This will make the proposed IDS apt for real-time applications in intricate ICPS environments.

Author Contribution Shareeful Islam: analysis and/or interpretation of data, security analysis, review and editing, research meetings, approved the final version, including references. Danish Javeed: security analysis, mathematical modeling, review and editing, research meetings, approved the final version including references. Muhammad Shahid Saeed: security analysis, mathematical modeling, review and editing, research meetings, approved the final version including references. Prabhat Kumar: prototyping, implementation, programming of deep learning technique, writing the original draft, review and editing, research meetings, approved the final submission including references. Alireza Jolfaei: supervision, validation, review and editing, research meetings, approved the final submission including references. AKM Najmul Islam: proposing the idea, supervision, conceptualization, review and editing, research meetings, approved the final submission including references.

Funding Open Access funding provided by LUT University (previously Lappeenranta University of Technology (LUT)). This work is supported by European Union's Horizon Europe research and innovation program under grant agreement No 101120779, CyberSecDome - An innovative Virtual Reality based intrusion detection, incident investigation and response approach for enhancing the resilience, security, privacy and accountability of complex and heterogeneous digital systems and infrastructures. This work was also partially supported by the Research Council of Finland with CHIST-ERA, grant agreement no - 359790, Di4SPDS-Distributed Intelligence for Enhancing Security and Privacy of Decentralised and Distributed Systems.

Data Availibility The datasets generated during and/or analyzed during the current study are available in the Kaggle, https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot and TON_IoT Datasets, https://research.unsw.edu.au/projects/toniot-datasets.

Declarations

Ethical Approval This article does not contain any studies with human participants and/or animals performed by any of the authors.

Competing Interest The authors declare no competing interests.

References

1. Yu X, Xue Y. Smart grids: a cyber-physical systems perspective. Proc IEEE. 2016;104(5):1058–70.
2. Kayan H, Nunes M, Rana OF, Burnap P, Perera C. Cybersecurity of Industrial Cyber-physical Systems: a review. ACM Comput Surv (CSUR). 2021;54:1–35.
3. Wright JG, Wolthusen SD. Access control and availability vulnerabilities in the iso/iec 61850 substation automation protocol. In Grigore Havarneanu, Roberto Setola, Hypatia Nassopoulos, and Stephen Wolthusen, editors, Critical Information Infrastructures Security, pages 239–251. Springer International Publishing. 2017.
4. Tidy J. How a ransomware attack cost one firm £45m. BBC; 2019. https://www.bbc.com/news/business-48661152. Accessed 10 Mar 1999.
5. Radiflow Team. Radiflow reveals first documented cryptocurrency malware attack on a SCADA network. radiflow; 2018. https://www.radiflow.com/news/radiflow-reveals-first-documented-cryptocurrency-malware-attack-on-a-scada-network/. Accessed 15 Mar 2023.

6. Islam S, Abba A, Ismail U, Mouratidis H, Papastergiou S. Vulnerability prediction for secure healthcare supply chain service delivery. Integr Comput-Aided Eng. 2022;29:1–21.
7. Kure H, Islam S, Mouratidis H. An integrated cyber security risk management framework and risk predication for the critical infrastructure protection. Neural Comput Appl. 2022;34:1–31.
8. Leander B, Causevic A, Hansson H. Applicability of the iec 62443 standard in industry 4.0 / iiot. In 14th International Conference on Availability, Reliability and Security. ACM. 2019.
9. Javeed D, Gao T, Saeed MS, Khan MT. Fog-empowered augmented intelligence-based proactive defensive mechanism for IoT-enabled smart industries. IEEE Internet Things J. 2023.
10. Alimi OA, Ouahada K, Abu-Mahfouz AM, Rimer S, Alimi KOA. A review of research works on supervised learning algorithms for SCADA intrusion detection and classification. Sustainability. 2021;13(17).
11. Bonagura V, Foglietta C, Panzieri S, Pascucci F. Advanced intrusion detection system for Industrial Cyber-Physical Systems. IFAC-PapersOnLine. 2022;55(40):265–270. 1st IFAC Workshop on Control of Complex Systems COSY 2022.
12. Althobaiti M, Kumar K, Gupta D, Kumar S, Mansour R. An intelligent cognitive computing based intrusion detection for Industrial Cyber-Physical Systems. Measurement. 2021;186:110145.
13. Althobaiti MM, Kumar KPM, Gupta D, Kumar S, Mansour RF. An intelligent cognitive computing based intrusion detection for Industrial Cyber-Physical Systems. Measurement. 2021;186.
14. Yaacoub J-PA, Salman O, Noura HN, Kaaniche N, Chehab A, Malli M. Cyber-physical systems security: limitations, issues and future trends. Microprocess Microsyst. 2020;77:103201.
15. Keshk M, Sitnikova E, Moustafa N, Hu J, Khalil I. An integrated framework for privacy-preserving based anomaly detection for cyber-physical systems. IEEE Trans Sustain Comput. 2019;6(1):66–79.
16. Jamal AA, Majid A-AM, Konev A, Kosachenko T, Shelupanov A. A review on security analysis of cyber physical systems using machine learning. Mater Today Proc. 2023;80:2302–2306.
17. El Houda ZA, Brik B, Khoukhi L. Why should I trust your IDS?: an explainable deep learning framework for intrusion detection systems in Internet of Things networks. IEEE Open J Commun Soc. 2022;3:1164–76.
18. Kayan H, Nunes M, Rana O, Burnap P, Perera C. Cybersecurity of Industrial Cyber-Physical Systems: a review. ACM Comput Surv (CSUR). 2022;54(11s):1–35.
19. Huo R, Zeng S, Wang Z, Shang J, Chen W, Huang T, Wang S, Yu FR, Liu Y. A comprehensive survey on blockchain in industrial Internet of Things: motivations, research progresses, and future challenges. IEEE Commun Surv Tutor. 2022;24(1):88–122.
20. Chae J, Lee S, Jang J, Hong S, Park K-J. A survey and perspective on Industrial Cyber-Physical Systems (ICPS): from ICPS to AI-augmented ICPS. IEEE Trans Industr Cyber-Phys Syst. 2023.
21. Lv Z, Chen D, Feng H, Singh AK, Wei W, Lv H. Computational intelligence in security of digital twins big graphic data in cyber-physical systems of smart cities. ACM Trans Manage Inf Syst (TMIS). 2022;13(4):1–17.
22. Gao Y, Chen J, Miao H, Song B, Lu Y, Pan W. Self-learning spatial distribution-based intrusion detection for Industrial Cyber-Physical Systems. IEEE Trans Comput Soc Syst. 2022;9(6):1693–702.
23. Alohali MA, Al-Wesabi FN, Hilal AM, Goel S, Gupta D, Khanna A. Artificial intelligence enabled intrusion detection systems for cognitive cyber-physical systems in industry 4.0 environment. Cogn Neurodyn. 2022;16(5):1045–1057.
24. Heng L, Weise T. Intrusion detection system using convolutional neuronal networks: a cognitive computing approach for anomaly detection based on deep learning. In 2019 IEEE 18th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC). 2019;34–40. IEEE.
25. Xuan CD, Huong DT, Nguyen T. A novel intelligent cognitive computing-based apt malware detection for endpoint systems. J Intell Fuzzy Syst. 2022;43(3):3527–47.
26. Schiliro F, Moustafa N, Razzak I, Beheshti A. Deepcog: a trustworthy deep learning-based human cognitive privacy framework in industrial policing. IEEE Trans Intell Transp Syst. 2022.
27. Fang W, Xue F, Ding Y, Xiong N, Leung VCM. Edgeke: an on-demand deep learning IoT system for cognitive big data on industrial edge devices. IEEE Trans Industr Inf. 2020;17(9): 6144–52.
28. Abdullahi M, Alhussian H, Aziz N, Abdulkadir SJ, Baashar Y. Deep learning model for cybersecurity attack detection in cyber-physical systems. In 2022 6th International Conference On Computing, Communication, Control And Automation (ICCUBEA). 2022;1–5. IEEE.
29. Hilal AM, Al-Otaibi S, Mahgoub H, Al-Wesabi FN, Aldehim G, Motwakel A, Rizwanullah M, Yaseen I. Deep learning enabled class imbalance with sand piper optimization based intrusion detection for secure cyber physical systems. Clust Comput. 2023;26(3):2085–98.
30. Sakhnini J, Karimipour H, Dehghantanha A, Yazdinejad A, Gadekallu TR, Victor N, Islam A. A generalizable deep neural network method for detecting attacks in Industrial Cyber-Physical Systems. IEEE Syst J. 2023.
31. Wang Z, Li Z, He D, Chan S. A lightweight approach for network intrusion detection in Industrial Cyber-Physical Systems based on knowledge distillation and deep metric learning. Expert Syst Appl. 2022;206.
32. Hossain MdD, Inoue H, Ochiai H, Fall D, Kadobayashi Y. Lstm-based intrusion detection system for in-vehicle can bus communications. Ieee Access. 2020;8:185489–185502.
33. Assis MVO, Carvalho LF, Lloret J, Proença ML Jr. A GRU deep learning system against attacks in software defined networks. J Netw Comput Appl. 2021;177.
34. Moustafa N. A new distributed architecture for evaluating AI-based security systems at the edge: network ton_IoT datasets. Sustain Cities Soc. 2021;72.
35. Ferrag MA, Friha O, Hamouda D, Maglaras L, Janicke H. Edge-IIotset: a new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access. 2022;10:40281–306.
36. Kumar P, Kumar R, Aljuhani A, Javeed D, Jolfaei A, Islam AKMN. Digital twin-driven SDN for smart grid: a deep learning integrated blockchain for cybersecurity. Solar Energy. 2023;263.
37. Javeed D, Gao T, Saeed MS, Kumar P, Kumar R, Jolfaei A. A softwarized intrusion detection system for IoT-enabled smart healthcare system. ACM Trans Internet Technol. 2023.

Authors and Affiliations

Shareeful Islam[1] · Danish Javeed[2] · Muhammad Shahid Saeed[3] · Prabhat Kumar[4] · Alireza Jolfaei[5] · A.K.M. Najmul Islam[4]

✉ Shareeful Islam
shareeful.islam@aru.ac.uk

✉ Prabhat Kumar
prabhat.kumar@lut.fi

Danish Javeed
2027016@stu.neu.edu.cn

Muhammad Shahid Saeed
ShahidSaeedRana@mail.dlut.edu.cn

Alireza Jolfaei
alireza.jolfaei@flinders.edu.au

A.K.M. Najmul Islam
najmul.islam@lut.fi

1 School of Computing and Information Science, Anglia Ruskin University, Cambridge, UK

2 Software College, Northeastern University, Shenyang 110169, China

3 School of Software Technology, Dalian University of Technology (DUT), Dalian 116024, Liaoning, China

4 Department of Software Engineering, LUT School of Engineering Science, LUT University, 53850 Lappeenranta, Finland

5 College of Science and Engineering, Flinders University, Adelaide, Australia