An Optimal Hybrid Cascade Regional Convolutional Network for Cyber Attack Detection

Ali Alqahtani[1], Surbhi Bhatia[2]*

[1]Department of Networks and Communications Engineering, College of Computer Science and Information Systems, Najran University, Najran 61441, Saudi Arabia,

[2]Department of Data Science, School of Science, Engineering and Environment, University of Salford, United Kingdom

Correspondence: surbhibhatia1988@yahoo.com

ABSTRACT

Cyber-Physical Systems (CPS) and the Internet of Things (IoT) technologies links urban systems through networks and improve the delivery of quality services to residents. To enhance municipality services, Information and Communication Technologies (ICTs) are integrated with urban systems. However, the large number of sensors in a smart city generates a significant amount of delicate data, like medical records, credit card numerics and location coordinates, which are transported across a network to data centers for analysis and processing. This makes smart cities vulnerable to cyberattacks due to the resource constraints of their technology infrastructure. Applications for smart cities pose many security challenges, such as zero-day attacks resulting from exploiting weaknesses in various protocols. Therefore, this paper proposes an optimal hybrid cascade regional convolutional network (hybrid TS-Cascade R-CNN) to detect cyber attacks. The proposed model combines the hybrid transit-search approach with the Cascade Regional Convolutional Neural Network to create an optimal solution for cyber-attack detection. The cascade regional convolutional network uses a hybrid transit search algorithm to enhance the effectiveness of cyber-attack detection. By integrating these two approaches, the system can leverage both global traffic patterns and local indicators to improve accuracy of attack detection. During training process, the proposed model recognizes and classifies malicious input even in the presence of sophisticated attack. Finally, the experimental analysis is carried out for various attacks based on different metrics. The accuracy rate attained by the proposed approach is 99.2%, which is acceptable according to standards.

Keywords:Cascade regional convolutional network, transit search algorithm, cyber attack, traffic, accuracy

1. Introduction

Artificial Intelligence (AI) techniques can provide a solution to this problem. AI refers to the rapid development of computer-based techniques, applications, and research that imitates human intelligence [1-3]. AI algorithms can quickly identify malware in applications and take necessary

actions. Additionally, the vast amount of information produced by users can be processed using AI [4]. To detect cyberattacks, Machine Learning (ML) with encoding, threat extraction, and enhanced security detection features are required. Deep Learning (DL) is a more effective approach for identifying cybersecurity issues, which is a powerful ML method driven by AI. DL can effectively process information in cybersecurity datasets and withstand attacks. Therefore, many researchers have applied DL to address cybersecurity challenges [5].

Human life is segregated into the facts and artificiality as real and virtual environment due to the existence of an internet society. It is believed that the maximum population are influenced by the virtual entities and virtual environment thus, criticizing the digital community. Because of these influences and fanaticism , the digital world terrorism and crimes related to Cyber-attacks are taking its edge and increasing so fast. It is high time to apply some ethical rules for the digital community in order to protect the innocent people and their lives from intruders by applying some mechanism with strict security obligations in the real world. These cyber attacks can be considered as actions that can be taken to avoid the protective measures in the digital world. [26]. Detecting these attacks could be defined as finding population who are unautorizingly making use of computer systems and who have legal access to the computer but abuse their pretensions. The deliberate manipulation of digital entities such as laptops, computers and mobile phones, i.e. technology based entities and networks [34-35]. These all poses a warning on the network sources related to its integrity, availability, and confidentiality. The different fourteen types of attacks can be divided into five primary sections [27] as DoS/DDoS attacks, infiltration attacks, malware attacks, information-gathering attacks, and man-in-the-middle attacks [31-32].

With the increasing sophistication and frequency of cyber-attacks [29-30], it has become crucial to develop robust and efficient systems for detecting and mitigating such threats. Traditional methods of cyber-attack detection often fall short in dealing with complex and evolving attacks. To address these challenges, a hybrid TS-Cascade R-CNN model has been proposed as an advanced method for cyber-attack detection.

1.1. Motivation and Contributions

Cyber-attacks can be analyzed through several models. However, if the attack traffic is not sufficiently characterized and profiled, AI-based models have low true positives and high false positives resulting to threats. As a result, the effectiveness of real-time categorization is limited and lowers the protection. We proposed a hybrid TS-Cascade R-CNN model to overcome this problem, reduce FPRs and enhance threat analysis. The developed model automatically learns temporal information using TS and spatial features utilizing Cascade RCNN without the need for human involvement. To enhance the efficiency, the dataset's significant features are automatically selected by the Cascade RCNN model, while the irrelevant characteristics are diminished. The parallel computation used by the TS model enhances computation time when retaining sequence modeling. Therefore, while offering low FPR and high accuracy, the hybrid model (Cascade RCNN-TS) aids in real-time analysis improvement in the smart city. Thus, the developed model

enhances smart cities' performance in accurately detecting cyberattacks. The outcomes of the evaluation show how well our developed model works.

The paper's main contributions are as follows:

1. The hybrid TS-Cascade R-CNN model is proposed for helping to enhance the threat detection.
2. The designing of the Transit search (TS) algorithm has been analyzed usinghost stars and signal-to-noise ratio (SNR).
3. For improving the accuracy of detection, high-quality object detectors are integrated into Cascade R-CNN through conquering the issue of overfitting during training and inference quality mismatch.
4. The optimization of hyperparameters is performed by employing the TS algorithm.
5. The experimentations and simulations have been performed using the Edge-IIoTset cyber security dataset.

1.2 Paper organization

The remaining section of the paper is arranged as follows. The literature review is described in Section 2 followed by the section 3 which states the proposed methodology. Furthermore, the experimentation results are listed in Section 4 and the last section concludes.

2. Related Work

Huma et al. [6] conducted a study on cyberattack detection in Industrial Internet of Things (IIoT) systems. The proposed methodology states the Hybrid Deep Random Neural Network (HDRaNN). The researchers evaluated the performance of HDRaNN using UNSW-NB15 and DS2OS datasets and assessed its effectiveness using various metrics. The results demonstrated that the HDRaNN technique outperformed traditional Deep Learning (DL) methods in accurately classifying sixteen types of cyberattacks, achieving a high level of accuracy.

In contrast, Elsaeidy et al. [7] focused on identifying replay attacks in smart cities.. The study utilized a dataset specifically designed for smart cities and evaluated the CNN model's ability to classify behavior as normal or abnormal. However, the model faced challenges in accurately representing real-world replay attacks on smart city infrastructure.

Ashraf et al. [8] employed the IoTBot-IDS framework to identify botnets in smart city networks. The framework utilized statistical learning techniques, including the Beta Mixture Model (BMM) and a Correntropy model, to represent the typical behavior of IoT networks. However, the model did not leverage AI-based Intrusion Detection Systems (IDSs) with computational learning techniques.

Diro and Chilamkurti [9] developed an approach using deep learning for detecting cyberattacks in IoT. Their approach focused on preventing hazardous attacks by discovering invisible instances during training using compression and self-learning abilities. The researchers validated their method using the NSL-KDD dataset and various machine learning methods, demonstrating high accuracy (99.20%) and overall performance.

Simon et al. [10] introduced method that did not incorporate feature selection techniques with metaheuristic algorithms, which could potentially enhance its performance.

Ma et al. [11] conducted research on association between mental health with the digital world. It justifies that the impact on mental health may depend on several factors, including the nature and quantity of online interactions, the content consumed, and individual differences in personality and social support. Further investigation is necessary to fully understand the complex interplay of these factors and develop effective policies to promote responsible social media usage.

Mirrashid et al. [12] explored the potential effects of AI adoption on job displacement, with some arguing that new career opportunities will arise alongside AI advancements. Studies suggest that jobs requiring advanced problem-solving and interpersonal skills are less likely to be automated, while repetitive occupations like data entry or assembly line labor are more susceptible to AI-driven automation. However, the impact of AI on the labor market may vary across sectors and regions, potentially exacerbating existing disparities. Policymakers and companies will need to implement retraining programs and provide assistance to workers in industries affected by AI-driven changes.

FERRAG, M. A [13] have been boosted by recent developments in genome editing tools like CRISPR-Cas9. However, there are ethical and safety issues to consider when using these technologies, such as the possibility of unintentional genetic modifications and the misuse of gene editing. The ethical concerns of changing the human genome, the need for regulatory control, and the effects on genetic variety and social justice have all been the subject of research into the pros and disadvantages of gene editing. It will be crucial to weigh the benefits of gene editing against its hazards and ethical concerns as the field develops. Bawany et al. [14] presented the SEAL model, which is a secure and agile software-defined networking framework for networking and protecting smart cities. This model utilized three distinct filter types for different types of applications, and the dynamic threshold was computed in real time using proactive, active, and passive filters. SEAL can effectively identify and mitigate DDoS attacks on network resources and application servers but is more time-consuming.

However, the time complexity is high in this scheme. Table 1 illustrates the previous related research works done in the field.

Table 1. Summary of the Exiting work performed by various researchers on this topic

Author	Technique	Application	Key finding	Limitation

Huma et al. [6]	Hybrid Deep Random Neural Network (HDRaNN).	IIoT.	High accuracy.	Requires lots of time for training.
Elsaeidy et al. [7]	convolutional neural network (CNN).	Replay attacks in smart cities.	The developed model is highly accurate in differentiating between normal and attack behaviors.	Real-world replay attacks on smart city infrastructure were not adequately portrayed
Ashraf, et al. [8]	Internet Of Things Botnet Intrusion Detection Systems (IoTBoT-IDS).	Botnets from Smart City.	Obtained accuracy of 99.2%.	ML and DL-based IDS detection algorithms' AI application was insufficient.
Diro and Chilamkurti et al. [9]	Distributed Deep-Learning.	IoT.	Achieves accuracy of 99.20%.	Obtains more time for learning.
Simon, J et al [10]	Convolutional Neural Network (CNN) and Decision Tree (DT).	IoT.	Maximum accuracy.	The developed approach wasn't used in feature selection methods utilizing metaheuristic algorithms.
Bawany et al. [14]	Secure and Agile (SEAL).	Networking and protecting smart cities.	Effectively identify and mitigate DDoS attacks.	More time-consuming.
Kumar et al. [15]	Privacy-Preserving and Secure Framework.	smart cities	Better performance.	Time complexity.
Memos, V.A. et al. [16]	Media-based Surveillance System	IoT network.	Shows better performance.	Requires lots of energy.
Xu, C. et al. [17]	DDoS attack Defense approach based on Traffic Classification (DDTC)	SDN-enabled smart cities.	accurately detect DDoS attacks as well as other unfamiliar attacks.	time complexity

Despite the current methods utilized for cyber-attack detection relying on independently scattered data samples, they still face difficulties in scaling to real-time applications due to their high computational costs. To address these drawbacks, a hybrid TS-Cascade R-CNN model is proposed in this research for the IoT network, which enhances the accuracy of cyber-attack detection by utilizing the advantages of deep learning. This approach can provide improved detection accuracy and shorter processing times, enabling it to adapt to the dynamic environment in milliseconds. A

defender's ability to discover and recognize cyber-attacks send back the arms race nature of the cyber domain. In recent years, defenders are developing new and improved techniques to detect known attacks, perform their intrusions, and detection of evade attackers using more sophisticated and stealthy techniques. Almost the existing datasets are older and may be inadequate to understand the latest behavior patterns of several cyber-attacks. Therefore, additional processing to make the target decisions yields low accuracy. The major challenge of data science in cybersecurity domain is finding a wide dataset on the risk prediction/ intrusion detection. The datasets may be noisy and full or errors or may contain some arbitrary instances related to security aspect which may directly affect the efficiency and performance of the models.

Major improvements

Several significant improvements have been made to enhance the efficiency and effectiveness of this model. By selectively attending to relevant information this model can achieve higher detection accuracy and reduce false positives. During training, this model learns to recognize and classify malicious inputs more accurately, even in the presence of sophisticated attack attempts designed to deceive the detection system. The system can dynamically adapt to changes in network configurations, such as the addition or removal of network nodes or services. This adaptability ensures that the detection system is effective in dynamic and evolving network infrastructure. The scalability of the model ensures that the system can deal with high-volume network traffic and maintain real-time detection capabilities in large-scale network environments.

3. Proposed Optimal Hybrid Cascade Regional Convolutional Network based on Attack Detection

In this research we have establishing the necessary hardware and software equipment for the study. A critical step in the process was to model threats and attacks against IIoT and IoT applications. Fourteen attacks associated with IIoT and IoT protocols were examined and categorized into five groups: DoS/DDoS attacks, infiltration attacks, malware attacks, information-gathering attacks, and man-in-the-middle attacks.Among these, DoS/DDoS attacks were identified as common types of attacks against IoT systems. They involve blocking valid requests from reaching the victim's IoT edge server by transmitting modified packets. The study specifically addressed four types of DoS/DDoS attacks: UDP Flood DDoS attack, ICMP Flood DDoS attack, TCP SYN Flood DDoS attack, and HTTP Flood DDoS attack.

The analysis of IOT data packets is done in order to explore vulnerabilities in the dataset. These attacks can be categorized as information-gathering attacks. They are known as OS fingerprinting, vulnerability scanning attacks, and port scanning. They can compromise mainly two entities which are IoT devices and edge servers. Man-in-the-middle attacks intercept communications between edge servers and IoT devices. The study examined two types of man-in-the-middle attacks: DNS spoofing attack and ARP spoofing attack.Malware attacks, including Password cracking attacks, Ransomware attacks, and Backdoor attacks, were also explored [13]. These attacks aim to gain

control over vulnerable components within the IoT network by installing backdoors. Another type of attack studied was injection attacks, where a malicious script is sent to an unknown, allowing access to sensitive data such as cookies and session tokens.

Following the attack modeling phase, the study proceeded to the data preprocessing stage, where features were extracted from the source data using an effective R-CNN (Region Convolutional Neural Network). The Faster R CNN is used as it can be trained at node to node. It is time efficient than state of the are traditional algorithms like Selective Search. Real-time classification of the threat type was achieved using the Cascade R-CNN technique. To enhance detection accuracy, high-quality object detectors were integrated into Cascade R-CNN, helping to overcome problems such as overfitting and inference quality mismatch. Figure 1 depicts framework model of the hybrid cascade R-CNN. The TS algorithm, renowned for its effectiveness in exoplanet exploration, was employed to optimize the hyperparameters of Cascade R-CNN. The best solution was determined by evaluating the fitness of each host star. The architecture of the proposed system is illustrated in Figure 2.

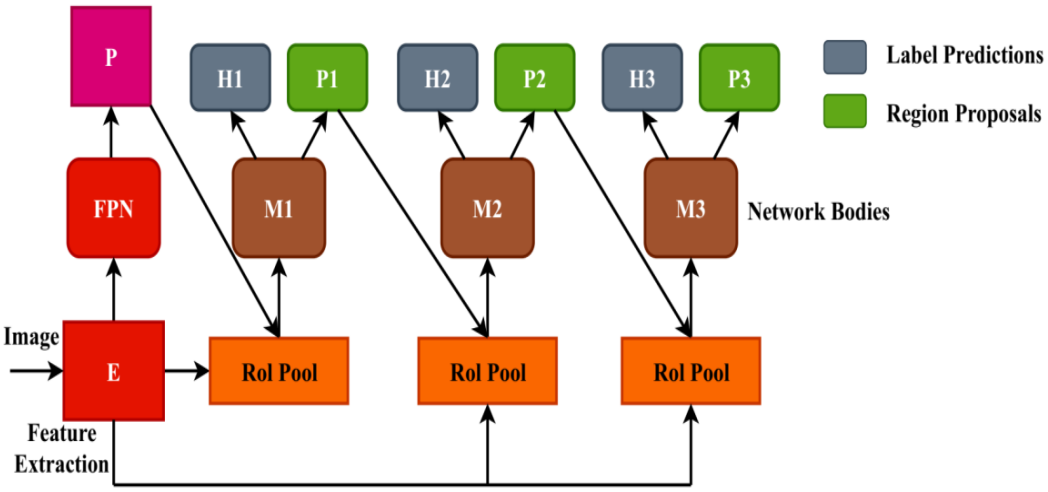

Figure 1: Framework model

The proposed model helps in detection as it utilizes the hybrid model in TS-cascade R-CNN is employed for anomalies and cyber attacks of intrusion in smart cities. The utilization of the hybrid model helps for the accurate detection because of the Edge-IIoT dataset. The hybrid T's cascade R-CNN is obtained to provide accurate information regarding cyber security attacks. In this model, the cascade is integrated which generates an instance segmentation for detection of attacks in IoT. The hybrid cascade is significant for the detection of different attacks. The association of R-CNN and TS cascade detects the cyber-attacksis determined in equation (1).

$$Y_n^b = \rho\left(Y, s_{n-1}\right), s_n = D_n\left(Y_n^b\right) \qquad (1)$$

Where the features are indicated by Y, the pooling operator is denoted by $\rho(\cdot)$, cascaded prediction is represented by s_n.

As mentioned in figure 2, the attacks based on IIoT as well as IoT applications such as Dos attack, DDoS attacks, Infiltration attacks, malware attacks, information gathering attacks as well as man in the middle attacks are determined from the provided database. The data determined are pre-processed to eliminate unwanted signals, data, noises, etc. The pre-processed data is then extracted and classified via optimal hybrid cascade R-CNN to attain predicted results.

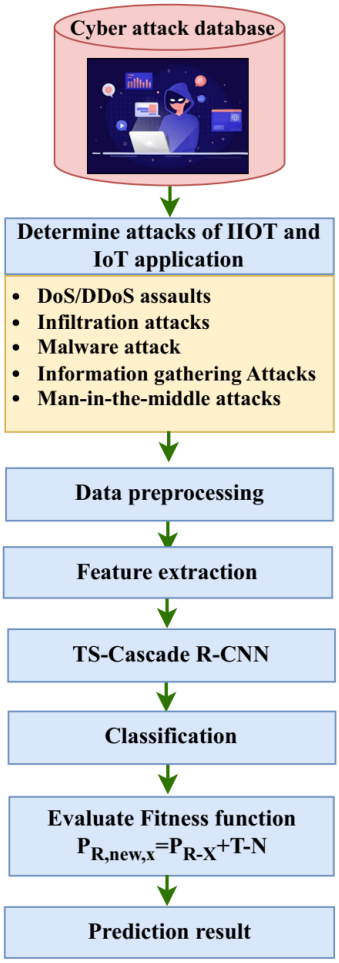

Figure 2: Flow chart for the proposed TS-Cascade R-CNN

3.1. Cyberattack model

The threats and attacks are modeled against IIoT and IoT applications in this stage. Fourteen attacks are exactly detected and evaluated. These are classified into 5 threats as information gathering, malware attacks, DDoS/DoS attacks, man-in-the-middle attacks, and injection attacks. The attacks are done by transmitting manipulated packets that formulate the IoT edge server of the victim engaged for legitimate requests. The proposed hybrid TS-Cascade R-CNN model architecture is illustrated in Figure 3, and the Edge-IIoTset cyber security dataset of IoT and IIoT was used as input.

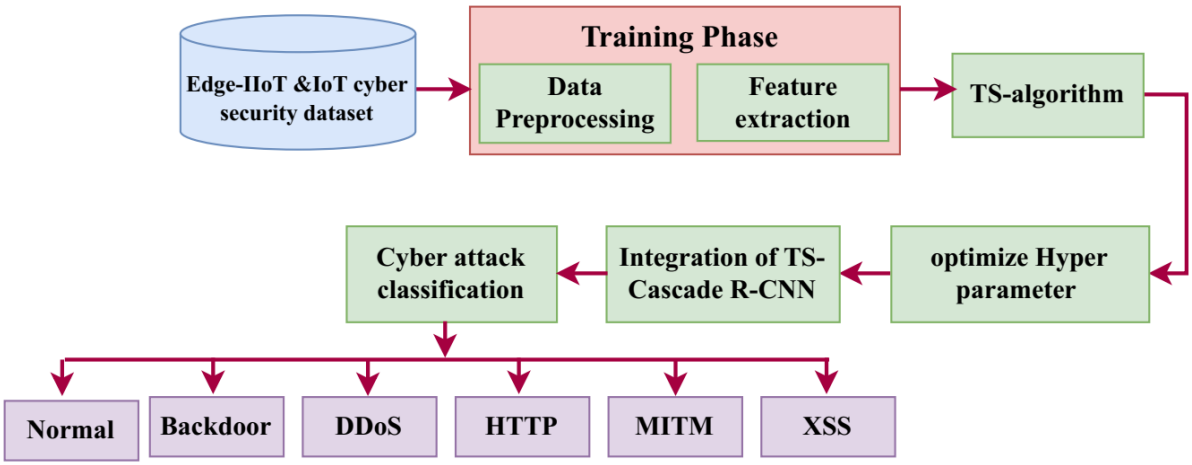

Figure 3. Proposed hybrid architecture

The data packets of IoT are examined in the information gathering for detecting the inadequacy of edge servers and IoT devices. It consists of 3 attacks such as vulnerability scanning attack, port scanning, and OS fingerprinting. The communications interception among edge servers and IoT devices are contained in the man-in-the-middle attacks, including DNS spoofing and ARP Spoofing attacks. The malicious script is transmitted to the innocent user which is referred to as an injection attack. This may take cookies, sensitive data, session tokens, and so on. At last, for managing weak network components of IoT, the installation of the backdoor is involved in malware attacks. It contains 3 attacks Ransomware attack, a Backdoor attack, and a Password cracking attack. The Edge-IIoTset cyber security dataset containing the attack scenarios list is presented in Table 2.

Table 2. Edge-IIoTset cyber security dataset containing the attack scenarios list.

Threats	Type of Attack
Man-in-the-middle attacks [19]	DNS spoofing
	ARP Spoofing
DDoS/DoS attacks [20]	HTTP flood DDoS attack

	TCP SYN Flood DDoS attack
	ICMP flood DDoS attack
	UDP flood DDoS
Information gathering [21]	vulnerability scanning attack
	port scanning
	OS fingerprinting
Malware attacks [22]	Ransomware attack
	Backdoor attack
	Password cracking attack
Injection attacks [23]	Uploading attack
	Cross-site scripting (XSS attack)
	SQL injection

3.2. Modified Cascade R-CNN

Two categories of object detection techniques exist one-stage techniques and two-stage techniques [11]. Utilizing two stages, object detection is viewed as a bounding box regression and classification problem with many learning tasks. Here, the computational load for two-stage approaches is often high. Then, the neural network just needs to be traversed once for one-stage algorithms to estimate every one of the bounding boxes in a single run. Owing to their high computational efficiency one-stage algorithms have lately gained popularity. Here, the two-stage Cascade Region Convolutional Neural Network (Cascade R-CNN) method is enhanced to offer more accurate cyberattack detection. The ResNet-101 backbone and the feature pyramids network (FPN) form the foundation of the Cascade R-CNN for comparisons. The bounding boxes were transformed to vector data once DPSs were found, and the number of DPS was denoted by the number of enclosing boxes.

By overcoming the overfitting issue during training and the inference quality mismatch issue, high-quality object detectors are incorporated into Cascade R-CNN to increase detection accuracy. On the Edge-IIoTset Cyber Security dataset, it is found that Cascade R-CNN, which is built on the FPN backbone and ResNet-101, outperformed several one-stage detectors, and two-stage detectors, for example, YOLOv2, and Faster R-CNN. Here, a Cascade R-CNN with the FPN backbone and ResNeXt-101 are employed [24]. DPSs are found in a variety of positions and shapes in a remotely sensed image. ResNeXt-101 is upgraded to add a Deformed ConvNet v2 (DCNv2) layer in place of the convolutional layer to enhance the learning of deformable features. When learning a spatial offset, the grid sampling sites swim concerning the feature map due to the development of DCNv2 over DCNv1. DCNv1 endures with the issue of the irrelevant image. Here, DCNv2 is more effective than DCNv1 at focusing on important image regions because it adapts to the structure of an object. The four various scales' characteristics are extracted by ResNeXt-101+DCNv2. Features from higher levels are iteratively fused to the present level via the FPN. Four stages are utilized in the fused features including three detectors, and one Region Proposal Network (RPN). These are applied in the first step.

3.3. Transit Search (TS) algorithm for improving the accuracy of Modified Cascade R-CNN

The signal-to-noise ratio (SNR) and the number of host stars ((m_r)) are the 2 parameters considered for the design of the TS algorithm [12]. The transit method detects the metric. Also, the attained standard deviation outside a transmission is utilized to evaluate the noise. Practically, photons obtained from the images of the stars can contain noise. The initial population size for TS is equivalent to the 2 parameters' product SNR and (m_r). The TS algorithm's execution stages and their interactions are defined in the below subsections. Star, galaxy, neighbor, transit, exploitation, and planet are the stages involved in TS execution that are discussed as follows.

3.3.1. Galaxy stage

The galaxy is chosen to initiate the algorithm. The galaxy center is the random location selected within a search space [25]. The galaxy's residential zones (life belts) have to be detected after the location selection. The locations where the life host has a high possibility are detected by calculating $m_r * SNR$, random locations P_{rand} using three equations. At last, choosing the containing optimal fitness. Chosen **l**ocations are likely to host life, and the further stages are initiated using these locations in an algorithm.

$$P_{rand,p} = P_{galax} + E - N \quad where, p = 1,....,(m_r \times SNR) \tag{2}$$

$$E = \begin{cases} v_1 P_{galax} - P_{rand} & if\ w = 1(negative location) \\ v_1 P_{galax} + P_{rand} & if\ w = 2(positive location) \end{cases} \tag{3}$$

The galaxy location is denoted as P_{galax}, and the random regions within the search space are indicated as P_{rand}. The random numbers within the range 0 and 1 are represented as coefficients v_1 and v_2 are the random vector that has several variables size in support of the optimization issue. The difference between the center of the galaxy and the random location describes the parameter E. The zone parameter w is the random number equivalent to 2 or 1. The noise is denoted as N which is utilized for improving the location accuracy. From every chosen location, the star that associates with the stellar system is selected in the next stage utilizing below three equations.

$$P_{R,x} = P_{rand,x} + E - N \quad where, x = 1,...., m_r \tag{4}$$

$$E = \begin{cases} v_4 P_{rand,x} - v_3 P_{rand} & if\ w = 1(negative location) \\ v_4 P_{rand,x} + v_3 P_{rand} & if\ w = 2(positive location) \end{cases} \tag{5}$$

$$N = (v_5)^3 P_{rand} \tag{6}$$

P_R is the stars' location in the above equation? Previous to the beginning of the iterations, the galaxy stage is performed in the algorithm. The random numbers within the interval $[0,1]$ are represented as v_3, and v_4, and the random vector is v_5.

3.3.2. Transit phase:

It is significant to recalculate the light received from the star to check for any possible reduction in the obtained light signals may help to identify the transit [2]. The corresponding fitness (e_R) P_R contain two meanings $N_1 and\ N_2$ in the TS algorithm. N_1 is employed when it is desired to determine and upgrade the planet's location using the location of the star. N_2 is employed when it is desired to ascertain and upgrade the brightness obtained from the star. Accordingly, a change P_R in the case N_2 denotes a new specification of the light signal, whereas a change P_R in the case N_1 denotes a change in the star's location. In addition, how bright a star seems to observers is known as apparent brightness. Stars are divided into seven major groupings in astronomy based on their luminosity and temperature. Blue light presents in the Largest and brightest stars. In this group, the star's Luminosity is more than 30,000 times of the sun. Additionally, red light from the faintest stars has a luminosity of less than 8% of that of the sun and has substantially, more mass, luminosity, and lower temperatures than the first group of stars. To calculate the habitat zone of stars, Luminosity is a significant parameter. Therefore, scientists take into account classes to discover planets where life is likely to exist near the star.

Determining the star classes is an important part of the TS algorithm. Therefore, utilizing the definition of N_2 the brightness of each star is examined. The small distance causes more protons to acquire. The below equation approximately obtains the star luminosity of the proposed algorithm.

$$P_x = \frac{Q_x / m_r}{(g_x)^2} \quad Q_1 \in \{1,2,.....,m_r\}\ x = 1,2,.....,mr \tag{7}$$

$$g_x = \sqrt{(P_R - P_D)^2} \quad x = 1,2,.....,mr \tag{8}$$

Where P_x and Q_x represents Luminosity and star rank respectively. Additionally, g_x addresses the separation between the telescope and the star x. At the beginning of the procedure, a random location for the telescope, P_D , is chosen; this location remains constant throughout the optimization. By modifying the value of P_R utilizing the definition N_2, the new signal is obtained to update the light that was received from the star. The coefficients v_6 and v_7, respectively, are random vectors with values between 0 and 1 and a number between -1 and 1.

$$P_{R,new,x} = P_{R,x} + T - N \quad x = 1,2,.....,mr \tag{9}$$

$$T = v_6 P_{R,x} \tag{10}$$

$$N = (v_7)^3 P_R \tag{11}$$

The amount of the star's brightness is computed, and the amount of the star's new luminosity $P_{x,new}$ is established by the below equation

$$P_{x,new} = \frac{Q_{x,new} / m_r}{(g_{x,new})^2} \quad x = 1,2,.....,mr \tag{12}$$

Using the new P_R parameters $g_{x,new}$ can be computed and also the telescope location. The possibility of transit can be defined by comparing P_x and $P_{x,new}$. P_D probability is represented by one and zero is described depending on the below equation. The planet phase is utilized if P_D =1 else implemented the neighbor phase in the present iteration.

$$if \;\; P_{x,new} < P_x \quad P_D = 1 \; (tranist) \tag{13}$$

$$if \;\; P_{x,new} < P_x \quad P_D = 0 \;\; (notransmit) \tag{14}$$

3.3.3. Planet phase:

In the previous phase, the value P_D is specified, in case the transit is noted (P_D =1), then in the TS algorithm, the planet phase is implemented. In this stage, initially, the first location of the identified planet is defined. from the star, the received light by the observer is obtained, so reduce the amount of light is appear while the planet is passed between the telescope and the star. Depending on this the first location of the identified planet P_w can be defined. This is done in the below equation of the TS algorithm.

$$x = 1,2,.....m_r \;\; P_w = (v_8 P_D + Q_P P_{R,x}) / 2 \tag{15}$$

Where

$$Q_P = P_{R,new,x} / P_{R,x} \tag{16}$$

The amount of signals received is analyzed to tell the location of the planet in star system by estimating the roundabout planet's situation. For this reason, the TS algorithm takes a variety of

SN signals. In the below equation, the coefficient v9 is a rand value between -1 and 1. Additionally, the random vector v10 has values between -1 and 1. After defining the signals P_n, tells the area of the planet, P_L. By adopting the mean SN signals utilizing the below equation

$$P_{n,y} = \begin{cases} P_w + v_9 P_q \ \ if \ \ w = 1 & for Aphelion\ region \\ P_w - v_9 P_q \ \ if \ \ w = 2 \quad y = 1,.....RM & for\ Perihelion\ region \\ P_w + v_{10} P_q \ \ if \ \ w = 3 & for\ \ Neutral\ region \end{cases} \tag{17}$$

$$P_L = \frac{\sum_{y=1}^{RM} P_{n,y}}{RM} \tag{18}$$

The farthest distance of the planet from the sun is called Aphelion and the closest distance is called perihelion. The TS algorithm considers the orbital position of the planet by three zones such as perihelion, neural region, and Aphelion is examined and pretentious by applying the zone parameter(w) in the phase of the planet. A random number 1,2, or 3 is a value parameter. For each mr star, there is only one planet in the TS algorithm.

3.3.4. Neighbor phase

Using Equation (18) to Equation (20), this is accomplished at the neighbor phase of the TS algorithm. First Equation (19) is used to determine the neighbor's initial location (P_w) while considering its host star $(P_{r,new})$ and a random location (P_{rand}). Using equations (20) and (21) the final location of the neighbor planet (N_M) is determined. A random number between 0 to 1 is dealt with by coefficients v_{11} and v_{12}. Additionally, the coefficients v_{13}, v_{14} in equation (19) are a random number and vector respectively between -1 and 1.

$$P_w = (v_{11} P_{r,new} + v_{12} P_{rand}) / 2 \tag{19}$$

$$P_{my} = \begin{cases} P_w - v_{13} P_{rand} \ \ if \ w = 1 & for\ Aphelion\ region \\ P_w + v_{13} P_{rand} \ \ if \ w = 2 \quad y = 1, \ldots, RM & for\ Perihelion\ region \\ P_w + v_{14} P_{rand} \ \ if \ w = 3 & for\ Neutral\ region \end{cases} \tag{20}$$

$$P_{M,i} = \frac{\sum_{y=1}^{RM} P_{m,y}}{RM} \tag{21}$$

The previous steps applied are regressed and are used to implement resampling in the subsequent stages. By Interaction Over Union (IoU) criteria of 0.7, 0.6, and 0.5, these 3 detectors were trained to locate a decent group in the further step. The Cascade R-CNN contained a regressor and a classifier that were both tuned for the IoU threshold at every stage. Algorithm 1 depicts the classification of cyber security attacks based on TS-Cascade R-CNN. Various cyber attacks are

determined and classify only significant attack that are determined for gathering the information based on cyber security.

Algorithm 1: Pseudocode for TS-Cascade R-CNN based cyberattack detection

Algorithm 1: Pseudocode for TS-Cascade R-CNN based cyberattack detection

Input: Initializing the R-CNN and TS

Output: Determine better cyber attack classification

1. Begin
2. Pre-processed data
3. Transmit the attacks by manipulating the packets to IoT edge server
4. For accuracy validation and prediction $Y_n^b = \rho(Y, s_{n-1})$ and $s_n = D_n(Y_n^b)$
5. Determine Hybrid TS Cascade R-CNN for classification
6. Perform classification of different attacks
7. Obtain better classification
8. End

Computational Complexity

The testing of the proposed method is accomplished and depends on samples, inputs and how many neurons are there in every layer and what are the boundary vectors. For training the computational complexity is O (n4) and testing as O (n2). The computational complexity for the training the algorithm is determined as O (n4) and for testing it is formulated as O (n2) based on summation.

4. Result and Discussion

The proposed Hybrid TS-Cascade R-CNN model is implemented with the Python programming language in Jupyter Notebook software. Also, the method used the system of MacBook Air with an Intel Core i5 650 CPU, MacOS 10.14 Mojave OS, 3.2 GHz processor speed, and 8 GB RAM for training the method. The size of the kernel is fixed as three and the epoch value is set as ten. The training batch size is 128. The training is done with the Edge-IIoTset Cyber Security Dataset and the performance evaluation metrics defined in the following section. The underlying Cyber-attack detection models like HDRaNN, CNN, distributed DL and IoTBoT-IDS are chosen for comparing the effectiveness of cyber threat identification.

4.1. Performance Metrics

The utilization of IIoT and IoT technology in smart city applications has been instrumental in enhancing the quality and efficiency of various city functions, including transportation, energy distribution, pollution control, and healthcare systems. The Edge-IIoTset Cyber Security Dataset serves as the input feature for the smart system. The cloud computing layer plays a vital role in storing large amounts of data acting as a intermediate between layers.. The terminal layer collects

data from IoT devices such as sensors. However, due to limited resources and the presence of various protocols, leading to latency issues in cloud centers. The integration of IoT ensures the sustainability and risk protection of smart city networks, enabling the development of services like smart buildings, transportation systems, and efficient resource utilization such as water management and waste disposal.

To detect intrusions and cyber-attacks in industries and smart cities, deep learning algorithms are suitable. The proposed model incorporates the IoTBoT-IDS dataset for embedded cyber-attack detection and applies the hybrid Transit Search-Cascade Regional Convolutional Neural Network (TS-Cascade R-CNN) to identify various cyber security attacks in smart cities.

To evaluate the performance of the proposed model, several metrics were utilized, including precision, F-score, recall, false positive rate (FPR), and accuracy. Accuracy helps in providing an indication of the model's precision in threat classification. The metrics given below [33-34] in Table 5for the proposed method in terms of threat classification.

4.2. Hyperparameter Settings

Table 3 depicts the hypermeter parameter settings used for tuning the proposed method.

Table 3. Hyperparameter settings

Parameters	Ranges
Total number of iterations	100
Size of population	30
Kernel size	3
Total number of epoch	10
Batch size	128

4.2. Dataset

The Edge-IIoTset cybersecurity dataset is a valuable resource used in IoT and IIoT applications [13]. This dataset has been employed in machine learning intrusion detection systems using both federated and centralized learning modes. To create comprehensive testbeds for evaluating cybersecurity measures, a layered approach is adopted, encompassing various components such as IoT and IIoT perception layers, Fog and Edge computing layer, Software-Defined Networking layer, Blockchain Network layer, cloud computing layer, and Network functions virtualization layer. Each layer contributes to the overall security infrastructure.The dataset comprises diverse IoT devices, including ultrasonic sensors, flame sensors, heart rate sensors, soil moisture sensors, and more. These devices generate different types of IoT data, which are essential for analyzing and identifying potential security threats. A total of fourteen attacks related to IoT and IoT protocols are thoroughly examined, and are divided into 5 groups: information gathering attacks, DoS/DDoS attacks, injection attacks, man-in-the-middle attacks, and malware attacks. This

categorization enables a systematic understanding of the types of threats that can target IoT and IIoT systems.

After the attacks are analyzed and categorized, the evaluation are shown in table 4 and provides an integral information in detecting and reducing cybersecurity threats in IoT and IIoT environments.

Table 4. Details of normal instances in the Edge-IIoTset dataset.

IP of Edge server	IP of IoT node server	IP of the Access point	IoT device type	MQTT Topic	Data type
192.168.0.128	192.168.0.101	192.168.0.1	DHT11 sensor	Temperature and Humidity	Periodic
192.168.1.128	192.168.1.101	192.168.1.1	HC-SR04 Ultrasonic sensor	Distance	Periodic
192.168.2.116	192.168.2.194	192.168.2.1	pH-sensor PH4502C	Ph Value	Periodic
192.168.3.12	192.168.3.18	192.168.3.1	Heart Rate Sensor	Heart_Rate	Periodic
192.168.4.30	192.168.4.73	192.168.4.1	Water sensor	Water level	Periodic
192.168.5.46	192.168.5.47	192.168.5.1	IR receiver sensor	IR-receiver	Random
192.168.6.100	192.168.6.56	192.168.6.1	LM393 Sound Detection sensor	Sound_Sensor	Random
192.168.7.55	192.168.7.62	192.168.7.1	G1006-based Flame sensor	Flame sensor	Random
192.168.0.128 192.168.7.55	192.168.0.101 192.168.7.62	192.168.0.1 192.168.7.1	Modbus/TCP server	Modbus _topic	Random
192.168.8.104	192.168.8.163	192.168.8.1	Soil Moisture Sensor v1.2	Soil_Moisture	Random

4.3. Performance Analysis

To expose the efficiency of our proposed Hybrid TS-Cascade R-CNN model we equated it with the Cascade RCNN model for comparing the efficiencies [28]. Cyber attacks are crucial and the results are mentioned in Table 5 which notes that the proposed Hybrid TS-Cascade R-CNN model would be showed greater performance. The output of the table showed the proposed model has higher performance than the Cascade R-CNN model concerning accuracy, recall, Precision, Sensitivity, Specificity, Computational time, and F-score [29]. The mean training time for each epoch with a variation of 305 s while training. Besides, The classification of the test dataset in the proposed method is done more quickly than the Cascade R-CNN model with the 22 s variance. The proposed method showed its performance by augmenting its speed and accuracy and reducing

the false positive rate and time taken for computation purposes. The sensitivity and specificity values are also enhanced by the method. Thus it can be deployed in practical applications.

Table 5. Comparison of performance of the proposed method in Edge-IIoTset Cyber Security Dataset.

Method	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)	Avg. training time / Epoch(s)	Classification time (S)	Sensitivity	Specificity	Computational time(s)
Cascade RCNN Model	98.57	98.59	98.25	98.47	1225.2	250	95.3	91.81	840
Hybrid TS-Cascade R-CNN model	99.30	99.15	99.48	99.56	1120.5	228	98.9	99.45	735

Table 6 presents the performance comparison results of the proposed Hybrid TS-Cascade R-CNN in Edge-IIoTset Cyber Security Dataset model with other existing methods in multiclass cyber attack classification. The table shows that all methods achieved high performance, but our proposed method outperformed all the other existing methods. The Hybrid TS-Cascade R-CNN model achieved 99.20%, 99.01%, and 99.41% for accuracy, precision, and recall, respectively, while the F-measure was 9.72%. Moreover, the model obtained the highest sensitivity and specificity values of 98.99% and 99.72%, respectively.

In addition, the proposed model achieved a reduction in computational time by approximately 26% compared to the existing approaches. This was achieved by optimizing the computational process and using less time for the computational process. The computational time for the proposed model is minimal, and it takes only 735 seconds to predict an attack. Overall, the results demonstrate that the proposed Hybrid TS-Cascade R-CNN model is an effective approach for improving the accuracy and reducing the computational time for cyber-attack detection in IoT and IIoT applications. Table 7 depicts the results for attacks with metrics.

Table 6. Comparison showing the metrics

Method	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Specificity	Computational time(s)
HDRaNN	98.21	96.32	97.05	98.00	90.98	1096
Faster R-CNN	98.90	98.52	99.00	99.01	92.43	856
Mask R-CNN	98.42	96.14	96.47	95.24	91.34	931
Distributed DL	96.41	95.13	97.52	96.71	95.45	892

Hybrid TS-Cascade R-CNN model	99.20	99.01	99.41	99.72	99.72	735

Table 7. Performance evaluation table for various types of attacks

Types of attacks	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)	Specificity (%)
DNS spoofing	93.5	92.3	90.45	89.4	87.6
ARP Spoofing	97.2	96.5	94.2	92.5	91.4
HTTP flood DDoS attack	96.17	94.9	93.5	91.7	90.3
TCP SYN Flood DDoS attack	95.4	93.2	91.6	90.4	88.4
ICMP flood DDoS attack	97.9	95.4	93.2	92.3	90.5
UDP flood DDoS	94.5	92.7	91.4	89.9	87.6
vulnerability scanning attack	98.3	96.1	95.3	93.2	92.7
port scanning	97.5	95.3	93.2	91.7	89.3
OS fingerprinting	93.6	90.4	89.4	87.4	84.5
Ransomware attack	98.7	97.7	95.23	93.3	91.8
Backdoor attack	99.4	98.3	96.4	94.7	92.5
Password cracking attack	96.5	94.2	92.7	91.9	87.6
Uploading attack	94.3	91.7	90.1	88.4	85.3
Cross-site scripting (XSS attack)	92.7	90.5	88.2	87.7	85.9
SQL injection	96.2	95.4	94.5	92.6	90.77

Figure 4 denotes that 97% of the attacks are classified precisely as attacks and the rest of the 3% alone are misclassified. Simultaneously, 98% of the normal data were categorized correctly as normal, and the remaining 2% were misclassified.

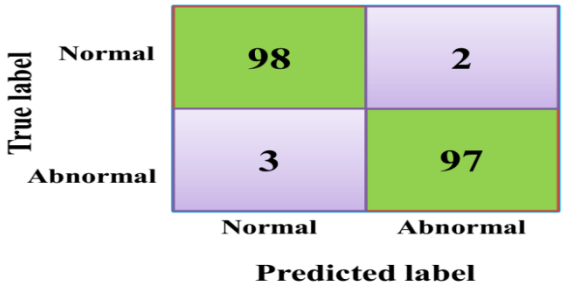

Figure 4. Confusion matrix based on the Edge-IIoTset Cyber Security Dataset.

Thus it explains that the proposed Hybrid TS-Cascade R-CNN model approach has significant performance in classifying cyber attacks. The effectiveness of the proposed model is defined in the Receiver Operating Characteristic (ROC) curve [30] and is represented in Figure 5. The ROC

curve values for all the attacks classification lay above 0.96% which reflects the quality of the classification method. The Macro average and micro average ROC curve values are 0.96 and 0.97.

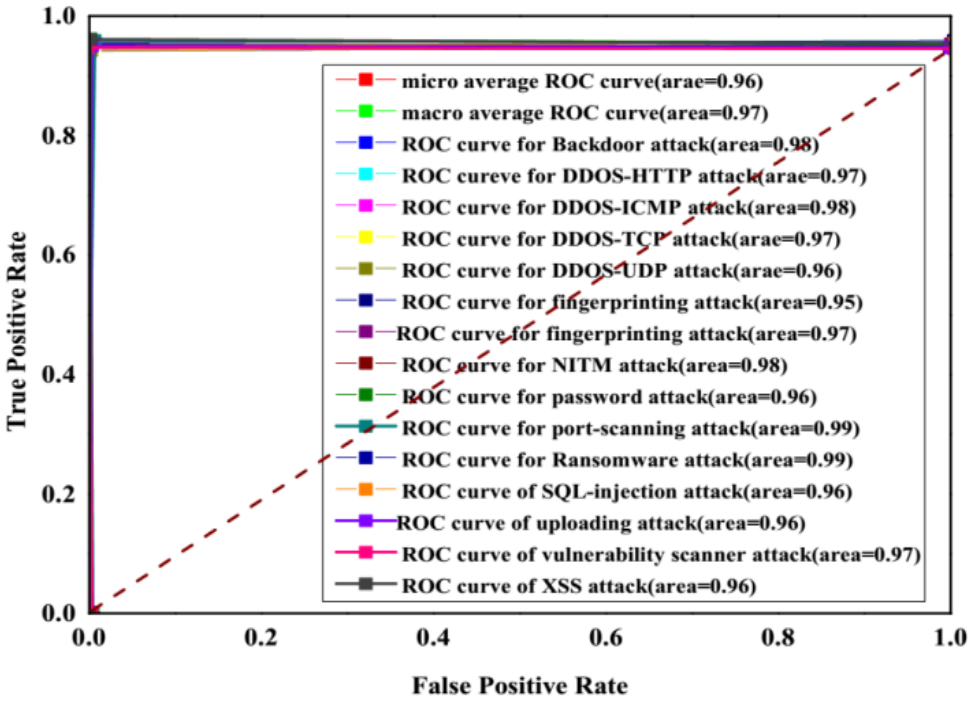

Figure 5. ROC curve for the Hybrid TS-Cascade R-CNN model.

Discussions

The problem of overfitting during training and inference quality mismatch issues have been resolved by using the retaining sequence modeling and the parallel computation used by the TS model improves computational time. The hybrid TS-Cascade R-CNN model is proposed to enhance the accuracy as listed in the results section in the Table 4. Although the current methods of cyber-attack detection rely on independently distributed data samples they face difficulties in scaling to real-time applications due to their high computational costs. This method can confer enhanced detection accuracy and shorter processing times, enabling adaptation to the dynamic environment in milliseconds. The study also evaluated three types of malware attacks, including Password cracking attacks, Ransomware attacks, and Backdoor attacks. The usage of the hybrid model in TS-cascade R-CNN is used for anomalies and cyber attacks of intrusion in a smart city. The utilization of the hybrid model helps for the accurate detection of cyber attacks in the smart city by using the Edge-IIoT dataset. By addressing the overfitting problem during training and the inference quality mismatch problem, higher-quality object detectors are integrated into Cascade R-CNN to maximize detection accuracy. In addition, the model shows the 26% computation time utilized and higher accuracy as compared with other traditional methods. The ROC curve values for all the attacks classification are above 0.96% which reflects the quality of the classification method. The Macro mean and micro mean ROC curve values are 0.96 and 0.97.

5. Conclusion

We have proposed a Hybrid TS-Cascade R-CNN model that demonstrates high efficacy in classifying cyber security threats. By combining the TS-cascade algorithm and RCNN methods, we were able to enhance accuracy and reduce the false positive rate values. Our model was implemented in Python and validated using the Edge-IIoTset Cyber Security Dataset. Comparison with four underlying methods, including HDRaNN, IoTBoT-IDS, distributed DL, and CNN, revealed the effectiveness of our proposed model. We computed performance metrics, such as accuracy, recall, precision, and F-measure values, which resulted in a significant increase in the classification time performance. The implementation of the TS-cascade algorithm led to a substantial reduction in computation time (1120.5s) and learning time (228s). The accuracy rate achieved by our model was 99.20%, while the macro and micro average ROC curves for multiclass classification of attacks were 0.97 and 0.96, respectively, indicating high classification efficiency. Furthermore, our model enables the classification of risk components into manageable risks, which are easy to measure, and uncontrollable risks, which are difficult to track. This classification system can aid in evaluating risks and identifying ways to mitigate them. In the future, we plan to develop our methodology further to identify additional attacks in IoT networks using our available resources.

Acknowledgement: This research was funded by Nazran University (Project number NU/DRP/SERC/12/41).

References

1. Rani, S., Kataria, A., Chauhan, M., Rattan, P., Kumar, R. and Sivaraman, A.K., 2022. Security and Privacy Challenges in the Deployment of Cyber-Physical Systems in Smart City Applications: State-of-Art Work. Materials Today: Proceedings.
2. Tao, H., Zain, J. M., Band, S. B., Sundaravadivazhagan, B., Mohamed, A., Marhoon, H. A., ... & Young, P. (2022). SDN-assisted technique for traffic control and information execution in vehicular adhoc networks. Computers and Electrical Engineering, 102, 108108.
3. Arikumar, K. S., Deepak Kumar, A., Gadekallu, T. R., Prathiba, S. B., & Tamilarasi, K. (2022). Real-Time 3D Object Detection and Classification in Autonomous Driving Environment Using 3D LiDAR and Camera Sensors. Electronics, 11(24), 4203.
4. Onyema, E. M., Kumar, M. A., Balasubaramanian, S., Bharany, S., Rehman, A. U., Eldin, E. T., & Shafiq, M. (2022). A security policy protocol for detection and prevention of internet control message protocol attacks in software defined networks. Sustainability, 14(19), 11950.
5. Dixit, P. and Silakari, S., 2021. Deep learning algorithms for cybersecurity applications: A technological and status review. Computer Science Review, 39, p.100317.
6. Huma, Z.E., Latif, S., Ahmad, J., Idrees, Z., Ibrar, A., Zou, Z., Alqahtani, F. and Baothman, F., 2021. A hybrid deep random neural network for cyberattack detection in the industrial internet of things. IEEE Access, 9, pp.55595-55605.
7. Elsaeidy, A.A., Jagannath, N., Sanchis, A.G., Jamalipour, A. and Munasinghe, K.S., 2020. Replay attack detection in smart cities using deep learning. IEEE Access, 8, pp.137825-137837.

8. Ashraf, J., Keshk, M., Moustafa, N., Abdel-Basset, M., Khurshid, H., Bakhshi, A.D. and Mostafa, R.R., 2021. IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities. Sustainable Cities and Society, 72, p.103041.
9. Diro, A.A. and Chilamkurti, N., 2018. Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, pp.761-768.
10. Simon, J., Kapileswar, N., Polasi, P.K. and Elaveini, M.A., 2022. Hybrid intrusion detection system for wireless IoT networks using deep learning algorithm. Computers and Electrical Engineering, 102, p.108190.
11. Saab Jr, S., Saab, K., Phoha, S., Zhu, M., & Ray, A. (2022). A multivariate adaptive gradient algorithm with reduced tuning efforts. Neural Networks, 152, 499-509.
12. Mirrashid, M. and Naderpour, H., 2022. Transit search: An optimization algorithm based on exoplanet exploration. Results in Control and Optimization, 7, p.100127.
13. FERRAG, M. A. (2022, March 18). Edge-iiotset cyber security dataset of IoT&iiot. Kaggle. Retrieved October 26, 2022, from https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot?select=Edge_IIoTset__DatasetFL.pdf
14. Bawany, N.Z. and Shamsi, J.A., 2019. SEAL: SDN-based secure and agile framework for protecting smart city applications from DDoS attacks. Journal of Network and Computer Applications, 145, p.102381.
15. Kumar, P., Kumar, R., Srivastava, G., Gupta, G.P., Tripathi, R., Gadekallu, T.R. and Xiong, N.N., 2021. PPSF: a privacy-preserving and secure framework using blockchain-based machine-learning for IoT-driven smart cities. IEEE Transactions on Network Science and Engineering, 8(3), pp.2326-2341.
16. Memos, V.A., Psannis, K.E., Ishibashi, Y., Kim, B.G. and Gupta, B.B., 2018. An efficient algorithm for media-based surveillance system (EAMSuS) in IoT smart city framework. Future Generation Computer Systems, 83, pp.619-628.
17. Xu, C., Lin, H., Wu, Y., Guo, X. and Lin, W., 2019. An SDNFV-based DDoS defense technology for smart cities. IEEE Access, 7, pp.137856-137874.
18. Ferrag, M.A., Friha, O., Hamouda, D., Maglaras, L. and Janicke, H., 2022. Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access, 10, pp.40281-40306.
19. Al-Shareeda, M.A. and Manickam, S., 2022. Man-in-the-middle attacks in mobile ad hoc networks (MANETs): Analysis and evaluation. Symmetry, 14(8), p.1543.
20. Almaraz-Rivera, J.G., Perez-Diaz, J.A. and Cantoral-Ceballos, J.A., 2022. Transport and Application Layer DDoS Attacks Detection to IoT Devices by Using Machine Learning and Deep Learning Models. Sensors, 22(9), p.3367.
21. Nissenbaum, H., 2020. Protecting privacy in an information age: The problem of privacy in public. In The Ethics of Information Technologies (pp. 141-178). Routledge.
22. Roseline, S.A. and Geetha, S., 2021. A comprehensive survey of tools and techniques mitigating computer and mobile malware attacks. Computers & Electrical Engineering, 92, p.107143.
23. Sah, D. K., Shivalingagowda, C., & Kumar, D. P. (2018). Optimization problems in wireless sensors networks. Soft computing in wireless sensor networks, 2018, 41-62..
24. Sun, P., Zhang, R., Jiang, Y., Kong, T., Xu, C., Zhan, W., Tomizuka, M., Li, L., Yuan, Z., Wang, C. and Luo, P., 2021. Sparse r-cnn: End-to-end object detection with learnable proposals.

In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 14454-14463).
25. Piccinni, O.J., Astone, P., D'Antonio, S., Frasca, S., Intini, G., La Rosa, I., Leaci, P., Mastrogiovanni, S., Miller, A. and Palomba, C., 2020. Directed search for continuous gravitational-wave signals from the Galactic Center in the Advanced LIGO second observing run. Physical Review D, 101(8), p.082004.
26. Zhang, J., Peng, S., Gao, Y., Zhang, Z., & Hong, Q. (2023). APMSA: Adversarial Perturbation Against Model Stealing Attacks. IEEE Transactions on Information Forensics and Security, 18. doi: 10.1109/TIFS.2023.3246766.
27. Li, B., Zhou, X., Ning, Z., Guan, X., & Yiu, K. C. (2022). Dynamic event-triggered security control for networked control systems with cyber-attacks: A model predictive control approach. Information Sciences, 612, 384-398. doi: https://doi.org/10.1016/j.ins.2022.08.093.
28. Balamurugan, V., Karthikeyan, R., Sundaravadivazhagan, B., & Cyriac, R. (2023). Enhanced Elman spike neural network based fractional order discrete Tchebyshev encryption fostered big data analytical method for enhancing cloud data security. Wireless Networks, 29(2), 523-537.
29. Lu, Z., Cheng, R., Jin, Y., Tan, K. C., & Deb, K. (2022). Neural Architecture Search as Multiobjective Optimization Benchmarks: Problem Formulation and Performance Assessment. IEEE Transactions on Evolutionary Computation. doi: 10.1109/TEVC.2022.3233364
30. Lu, H., Zhu, Y., Yin, M., Yin, G., & Xie, L. (2022). Multimodal Fusion Convolutional Neural Network With Cross-Attention Mechanism for Internal Defect Detection of Magnetic Tile. IEEE Access, 10, 60876-60886. doi: 10.1109/ACCESS.2022.3180725

31 Yu, J., Lu, L., Chen, Y., Zhu, Y., & Kong, L. (2021). An Indirect Eavesdropping Attack of Keystrokes on Touch Screen through Acoustic Sensing. IEEE Transactions on Mobile Computing, 20(2), 337-351. doi: 10.1109/TMC.2019.2947468
32. Qiao, F., Li, Z., & Kong, Y. (2023). A Privacy-Aware and Incremental Defense Method Against GAN-Based Poisoning Attack. IEEE Transactions on Computational Social Systems. doi: 10.1109/TCSS.2023.3263241
33. Saab Jr, S., Fu, Y., Ray, A., & Hauser, M. (2022). A dynamically stabilized recurrent neural network. Neural Processing Letters, 54(2), 1195-1209.
34. Han, Z., Yang, Y., Wang, W., Zhou, L., Gadekallu, T. R., Alazab, M., ... & Su, C. (2022). RSSI Map-Based Trajectory Design for UGV Against Malicious Radio Source: A Reinforcement Learning Approach. IEEE Transactions on Intelligent Transportation Systems
35. Donta, P. K., Amgoth, T., & Annavarapu, C. S. R. (2022). Delay-aware data fusion in duty-cycled wireless sensor networks: A Q-learning approach. Sustainable Computing: Informatics and Systems, 33, 100642.