ISSN: 2093-5374 / E-ISSN: 2093-5382

Cyber Attack Recognition in an Internet of Things-Enabled Environment Using a Hybrid Optimised Deep Learning Approach

Boyella Mala Konda Reddy[1*], Dr.A. Abdul Azeez Khan[2], Dr.K. Javubar Sathick[3], and Dr.L. Arun Raj[4]

[1*]Department of Computer Science and Engineering, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India. uboyella_cse@crescent.education, https://orcid.org/0009-0002-0894-9352

[2]Associate Professor, Department of Computer Applications, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India. abdulazeezkhan@crescent.education, https://orcid.org/0000-0001-6960-752X

[3]Associate Professor, Department of Computer Applications, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India. javubar@crescent.education, https://orcid.org/0000-0002-2248-8380

[4]Associate Professor, Department of Computer Science and Engineering, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India. arunraj@crescent.education, https://orcid.org/0000-0001-8181-5022

Received: October 17, 2024; Revised: December 05, 2024; Accepted: January 07, 2025; Published: March 31, 2025

Abstract

A cyber-attack is the malicious manipulation of computer networks and systems to compromise data or impede procedures and operations using malware. With the exponential growth in computational capacity, machine learning (ML) and deep learning (DL) approaches have emerged as promising countermeasures for advancing and identifying such threats. To address this challenge, a novel optimized deep hybrid attack detection model called SCEEHO-SPC-CNN-CD-DBN is proposed in this research article. Data is subjected to a preprocessing procedure before it is used for further processes. Here, the data undergoes a normalizing phase for pre-processing, during which the statistics and higher-order statistical features are retrieved. The cyber-attack detection process concludes with a hybrid DL model applied to the retrieved features. The proposed hybrid classifier integrates models such as the DBN (Deep Belief Network) with contrastive divergence (CD) and the split convolution module (SPC)-based CNN (Convolutional Neural Network). Training the CNN and DBN using the SCEEHO(Sea CrowEndorsed Elephant Herding optimization) model and fine-tuning the ideal weights improves detection accuracy. Furthermore, have tested the developedSCEEHO-SPC-CNN-CD-DBN-based hybrid classifier on the CIC IoT Dataset 2023. The evaluated results, employing a wide range of statistical measures, demonstrate that the research model performs efficiently.

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 16, number: 1 (March), pp. 49-71. DOI: *10.58346/JOWUA.2025.I1.003*

* Corresponding author: Department of Computer Science and Engineering, B. S. Abdur Rahman Crescent Institute of Science & Technology, Chennai, Tamil Nadu, India.

Keywords: Cyber-Attack, Deep Learning, Optimized Deep Hybrid Attack Detection, Convolutional Neural Network, Deep Belief Network, and Sea Crow Endorsed Elephant Herding Optimization.

1 Introduction

Physical devices of varied form factors and levels of computing, sensing, and communication power are the compositions of the Internet of Things (IoT) (Gubbi et al., 2013). Cyberspace and microchips allow these devices to communicate with one another, facilitating effortless two-way communication. Interconnected "things" such as the internet, indicators, and web-based communication as well as management technologies make human life easier (Madakam et al., 2015). The environment is monitored by microchips that relay data to humans and the internet (Husamuddin & Qayyum, 2017). As more and more people use computers and the internet in their daily lives, there is a corresponding rise in the need to protect their data and personal information. Recent attempts to infiltrate computer networks and systems have increased in combination with the explosion of Internet-based applications and the rise of advanced technology like the IoT (Mahmoud et al., 2015).

The IoT enables the networked connection of many objects equipped with sensors (including coffee makers, lights, smart TVs, refrigerators, and many more) in fields as varied as medicine, agriculture, transportation, and more (Perwej et al., 2019). Work and life are being revolutionized by Internet of Things applications that help save time and money. Its potential uses are endless, and it provides a lot of options for learning, development, and progress (Shanmugasundaram et al., 2023). Since the Internet serves as the backbone and main role of the IoT, every security risk that exists on the Internet also exists there. According to a survey from SonicWall (Den et al., 2021), cyber criminals in India increased their use of ransomware and IoT cyber-attacks in the first half of 2023. According to the 2023 Mid-Year Cyber Threat Report published by SonicWall, a publisher of cyberattack analytics and ransomware statistics, ransomware attacks in India increased by 133% while Internet-of-Things attacks increased by 311%. Nodes in the Internet of Things are underpowered, under-resourced, and uncontrollable by humans (Den et al., 2021; Ammi & Jama, 2023).

Network-based security solutions are needed because of the cumulative prevalence of IoT devices in daily life and the associated security risks. While existing systems can successfully identify certain types of attacks, others remain elusive. There is slight uncertainty that network security might benefit from more advanced approaches; as both the frequency and severity of attacks on networks and the volume of data stored in them continue to rise, faster and more effective attack detection methods are essential (Diwakar & Roy, 2024; Alsamiri & Alsubhi, 2019). ML is one of the most successful computing models in this context, and it is increasingly being used to give embedded intelligence in the IoT environment, as depicted in Figure 1. There have been many applications of ML to the problem of network security (Inayat et al., 2022), including analysis of network traffic (Churcher et al., 2021), detection of intrusion (Verma & Ranga, 2020), and identification of botnets (Bansal & Mahapatra, 2017; Muralidharan, 2024).

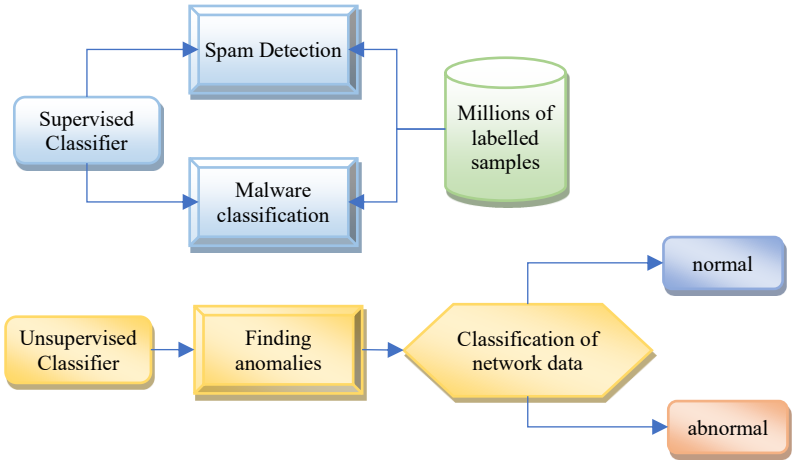

Figure1: Machine Learning Classifiers-Based Cyberattacks Detection

When it comes to cyber analysis, ML can be employed in two primary ways: signature-based (exploit-based) and anomaly-based (Manjramkar & Jondale, 2023). Signature-based approaches aim to identify known attacks based on unique features of the traffic used in those attacks (or "signatures") (Jeong et al., 2019). The ability to efficiently detect all known threats without producing an overwhelming number of negative results is a key benefit of this class of detection technology. Some current works in the literature use signature-based algorithms to detect attacks (Malik et al., 2022). For instance, four unique ML techniques were applied as exploratory tools in the field of network traffic analysis (Churcher et al., 2021), to learn the defining features of some well-known attacks. Classic ML, on the other hand, is not very good at automatically processing features has a low detection rate (Den et al., 2021; Sarhan et al., 2022) and also struggles to spot even minor changes in attacks. This has prompted the exploration of DL methods for enhancing cyber-security infrastructure (Alisawi et al., 2023).

By highlighting deviations from typical behaviour, DL methods help cyber-security schemes to prevent intrusions (Roopak et al., 2019; Alnumay, 2024). Novel methods for IDS are necessary (Verma & Ranga, 2020) notwithstanding the characteristics of IoT networks such as the distributed nature and inadequate computing competencies of end devices (Ahmed & Pandey, 2024). Frame detection and DL are used to prevent attacks on the IoT in this work. The index system for cybersecurity warning systems is built, index features are selected and restrained, and lastly, the situation is evaluated to launch a cybersecurity warning system in a big data set. Several swarm and bio-inspired methods were implemented to reduce the dimensionality of data and eliminate extraneous or noisy input, hence increasing an IDS's efficiency (Lotfy & Vatankhah, 2014; Yogamadhavan & Mannayee, 2024). An improved bio-inspired algorithm called SCEEHO has been developed to aid IoT intrusion detection by the IDS, where it is here used to optimize the hyperparameters of the hybrid deep learning model (Salman & Alomari, 2023). The following are some of the main contributions of the proposed work:

- In this research, a high-performance deep learning framework, SCEEHO-SPC-CNN-CD-DBN, is specifically designed to detect and classify cyber-attacks within IoT-enabled environments.
- This model integrates the advantages of both CNN and DBN models while optimizing their performance using the SCEEHO algorithm. The attack detection model is developed to enhance detection accuracy and robustness against diverse cyber threats (Sathish Kumar, 2024).
- The CIC IoT Dataset 2023 was preprocessed to extract statistical and higher-order features, crucial for the efficacy of the proposed detection framework. This preprocessing phase involves

normalization and feature extraction processes that significantly contribute to the model's enhanced performance, ensuring it accurately identifies and classifies various cyber threats.

- Hyperparameter tuning is conducted using the SCEEHO algorithm to optimize the performance of the deep learning models within the framework.
- The proposed SCEEHO-SPC-CNN-CD-DBN model is evaluated and validated against existing state-of-the-art methods to assess its effectiveness in detecting a broad spectrum of cyberattacks. The comparative analysis highlights the strengths and limitations of the proposed approach, demonstrating its superiority in terms of accuracy, efficiency, and adaptability in IoT-enabled environments.

This paper's remaining sections are organized as follows. The existing attack detection using DL in IoT networks is discussed in Section 2. The next section includes the dataset details and methodologies employed in the development of a distributed attack detection framework. The hyperparameter tuning of the proposed method is outlined in Section 4. The performance evaluation of the distributed attack detection framework is assessed in Section 5. The conclusion of the study and future works are presented in Section 6.

2 Related Work

This review analyzes the current research works developed for the detection and classification of cyber attacks that occur in IoT. Based on the review, the analyzed current research works are discussed with the research model’s pros and cons comparatively. Dey et al., (2023) introduced a hybrid feature selection methodology that integrates statistical test-based filter techniques, including Mutual Information (MI), Chi-Square (χ^2), and, PCC (Pearson's Correlation Coefficient)with a metaheuristic approach called NSGA-II (Non-Dominated Sorting Genetic Algorithm) for feature optimization. The performance assessment of the projected scheme is conducted using the ToN-IoT dataset, with a focus on two key metrics: the number of selected features and the accuracy achieved. Nevertheless, the development of a highly effective and expeditious IDS to identify and mitigate cyber-attacks remains a complex and unresolved area of academic investigation.

Abirami & Palanikumar (2023) employed a classification approach known as Deliberate Deep Reinforced Learning (DDRL) to make predictions for normal and attacking data, utilizing the relevant attributes. The reliable prediction of incursions from cyber-datasets is achieved by the utilization of the GRN(Graph Referencing Normalization), DDRL, and BBBC(Big-Bang Big-Crunch)methods. Nonetheless, the system encounters challenges such as the lack of dependable attack detection, complex scheme modelling, substantial computational expenses, and prolonged processing durations. Ding et al., (2023) proposed a novel DL framework called DeepAK-IoT to identify and mitigate cyberattacks against IoT devices. The system's limited attributes for describing attacks necessitate improvement through the development of more sophisticated detection criteria, hence enabling the creation of attack signatures.

Al-Haija et al., (2021) introduced an innovative autonomous system capable of identifying mutations in IoT cyber-attacks. This system utilizes a deep CNN for recognition purposes. Nevertheless, the viability of implementing such a detection methodology inside a resource-limited IoT setting raises doubts. Kandhro et al., (2023) presented a unique approach that utilizes DL techniques for the detection of security holes and holes in CPS (cyber-physical systems). The framework under consideration presents a comparison between unsupervised methods and DL-based selective approaches.

Nevertheless, the building and design of DL models with intensity explosion and extinction pose significant obstacles. Khoa et al., (2022) introduced an innovative collaborative learning paradigm that incorporates Transfer Learning (TL). This framework facilitates the efficient and rapid acquisition of "knowledge" by a target network, which lacks labelled data, from a source network that holds a substantial amount of labelled data. Nevertheless, a significant challenge in implementing Federated Learning (FL) within the IoT networks pertains to the absence of labelled data and the divergence in data characteristics for training purposes.

Abusnaina et al., (2021) proposed DL-FHMC, a method that employs fine-grained hierarchical learning to enhance the robustness of IoT malware detection (Gyamfi et al., 2022). This methodology exploits behavioural patterns derived from Control Flow Graph (CFG) analysis to identify and detect harmful adversarial IoT software. Nevertheless, subsequent studies have demonstrated the vulnerability of these methods to adversarial attacks, wherein perturbations are introduced into the feature space. Shahin et al., (2022) developed deep hybrid learning models that combined ALSTM(Attention-based Long Short-Term Memory) and FCN(Fully Convolutional Neural Network) with GBoost (Gradient Boosting) techniques, including AdaBoost (Adaptive Boost) and XGBoost (Extreme Gradient Boosting). These models were specifically designed to distinguish anomalies in traffic data originating from industrial IoT devices. The authors omitted any mention of potential avenues for future research and the current issues that arise as a result of the vulnerabilities and dangers they have highlighted.

Rajkumar et al., (2023) employed a DL approach that combines SLSTM (stacked long short-term memory) with pre-trained ML models. This strategy was utilized to effectively understand the characteristics of suspicious activities comprehensively and accurately distinguish them from regular traffic. Mehedi et al., (2022) utilized CNN and LSTM models, together with character encoding, to investigate the efficacy of SFL(Spatial Feature Learning) methods in differentiating complicated and encrypted HTTP data from ordinary traffic.

Table 1: Comparative Analysis of Reviewed Current Models

Ref	Methods Used	Application	Pros	Cons
(Dey et al., 2023)	MI, Chi-Square, and PCC with NSGA-II	Intrusion Detection in IoT using ToN-IoT dataset	Effective feature selection and improved accuracy.	Complexity in developing a highly effective and fast IDS; unresolved challenges in cyber-attack identification and mitigation.
(Abirami & Palanikumar, 2023)	DDRL with GRN and BBBC	Cyber-attack detection using relevant attributes	Reliable prediction of incursions and effective use of advanced ML methods.	High computational cost, complex modelling, long processing times, and lack of dependable attack detection.
(Gyamfi et al., 2022)	DeepAK-IoT	Cyber-attack detection in IoT devices	Novel DL framework for attack detection.	Limited attributes for attack description, requiring improved detection criteria and attack signatures.
(Al-Haija et al., 2021)	DeepCNN	Mutation identification in IoT cyber-attacks	Autonomous system capable of mutation detection.	Challenges of viability in resource-limited IoT environments.
(Kandhro et al., 2023)	DL techniques for detecting security holes in CPS	Security hole detection in CPS	Comparison of unsupervised and DL-based selective approaches.	Challenges with intensity explosion and extinction in DL models.
(Yogamadhavan & Mannayee, 2024)	Collaborative learning withTL	Knowledge transfer in IoT networks with labelled/unlabelled data	Efficient and rapid acquisition of knowledge from a source network.	Issues with implementing FL in IoT due to lack of labelled data and data characteristic divergence.

(Ahmed & Pandey, 2024)	DL-FHMC and CFG analysis	IoT malware detection using behavioural patterns	Enhances robustness in malware detection through behavioural analysis.	Vulnerable to adversarial attacks with feature space perturbations.
(Alisawi et al., 2023)	Deep hybrid models combining ALSTM, FCN with GBoost	Anomaly detection in industrial IoT traffic data	Effective anomaly detection combining multiple advanced ML techniques.	Lack of discussion on future research directions and unresolved issues related to vulnerabilities.
(Rajkumar et al., 2023)	SLSTM combined with pre-trained ML models	Characterization of suspicious activities in network traffic	Comprehensive understanding and accurate distinction of suspicious activities from regular traffic.	No specific disadvantages were mentioned, but the study has limitations in scalability and generalization.
(Lotfy & Vatankhah, 2014)	CNN and LSTM with character encoding for SFL	Differentiation of complex and encrypted HTTP data	Effective in distinguishing between complicated/encrypted HTTP data and regular traffic using SFL methods.	Requires significant computational resources and the approach might be less effective on non-HTTP traffic types.

In table 1, Several surveys have examined various approaches to designing cyber-attack detection systems for IoT networks. However, most of these surveys have not thoroughly explored the implementation and optimization of DL techniques as detection mechanisms for IoT networks and their lightweight devices (Suvarna & Bharadwaj, 2024). Several studies published in the range of 2016 to 2023 have examined the topic of IoT security, specifically focusing on issues and classification. These studies have provided an inclusive review of DL techniques for IDSs in IoT networks. However, further analysis and investigation are required, particularly in the area of hyperparameter tuning of DL methods, as this could enhance the accuracy of IDSs. Improving accuracy is a primary objective of the present study.

3 Proposed Methodology

The proposed research method, SCEEHO-SPC-CNN-CD-DBN, provides a significant advancement over earlier works by integrating a highly optimized and hybrid DLmodel for cyber-attacksidentification in IoT environment. Unlike previous models that often struggled with feature selection complexity, high computational costs, and limited detection capabilities, this research method utilizes the SCEEHO to fine-tune the parameters of both the CNN (with a Split Convolution Module) and the DBN with CD. This results in an enhanced capability to capture intricate patterns within the data, leading to superior accuracy in identifying diverse types of cyber-attacks. Additionally, by normalizing and preprocessing the CICIoT2023 dataset, the research model effectively addresses the limitations of prior approaches, such as inadequate feature representation and vulnerability to adversarial attacks. The hybrid model's ability to integrate and optimize multiple learning algorithms ensures robustness and efficiency, making it more effective in real-time and resource-constrained IoT environments compared to the earlier works.The research model, SCEEHO-SPC-CNN-CD-DBN is a novel deep learning framework designed to enhance the detection of cyber-attacks in IoT environments by employing the advantages of multiple machine learning components, each optimized for specific tasks within the overall detection process.

The research objective is to develop a unique deep hybrid cyber-attack detection model comprising two distinct phases: preprocessing and classification. The incoming data is initially subjected to two pre-processing techniques, namely feature selection and data processing. Additionally, the statistical characteristics are extracted. Additionally, the characteristics that have been retrieved are utilized as the

input for the detection phase, employing a hybrid model that integrates methods such as SPC-CNN and CD-DBN. The detection model must undergo appropriate training procedures to enhance its detection accuracy. To achieve this objective, this work incorporates the application of hybrid optimization methods to enhance the efficiency of the training process. Consequently, the optimal tuning of the weights for both the proposed DL models is achieved through the utilization of a novel SCEEHO algorithm. The proposed algorithm under consideration is a hybrid approach that integrates the principles and methodologies of both the EEHO and SCOA algorithms. Figure 2 depicts the architectural design of the detecting system that has been proposed.

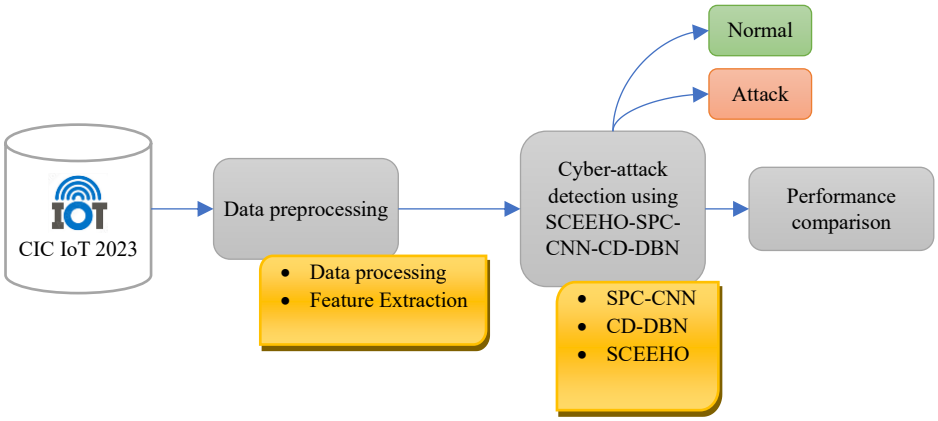

Figure 2: Proposed IoT Cyber-Attack Detection Model Architecture

1) Dataset Description

The most recent CIC IoT Dataset (2023) has been used in the proposed SCEEHO-SPC-CNN-CD-DBN effort. The primary purpose of the researchwas to utilizea comprehensive IoT attacks dataset to analyze the growth of safety analytics usage in operational IoT settings. This is done by carrying out 33 attacks on a 105-node IoT network. DoS, DDoS, Recon, Brute Force, Web-based, Spoofing, and Mirai are just a few of the names given to these various forms of cyberattack. Last, every single attack is carried out by hostile IoT devices that specifically target other IoT devices. This dataset includes the newest available network data, both with and without an attack, and is designed to be as representative as possible of the network of a working organization. Since the imbalance of this dataset has a significant impact on the training of the DL method and, by extension, the testing, furthermore have corrected the problem by creating duplicates of the data. Table 2 shows the total number of all the labels that encompass benign traffic. The dataset has 1 label and 46 features.

Table 2: CICIoT2023 Traffic Distribution

Traffic Types	Training	Testing
Benign	138542	39837
Recon-PortScan	11379	1952
Recon-OSScan	14965	980
Recon-HostDiscovery	18448	3286
Recon-PingSweep	335	2
Mirai-greeth_flood	130796	29478
Mirai-udpplain	115633	28787
Mirai-greip_flood	95402	27056

2) Data Pre-Processing

The purpose of the pre-processing of the CIC IoT Dataset is to standardize the raw data. For the next phase of processing to go smoothly. The objective is to train the machine-learning system to distinguish the attributes of attack features. The pre-processing phase consists of four sessions with the same end. Here is a detailed description of each step involved in the statistical pre-processing:

Deleting outlier: There is a requirement to remove a value from the CIC IoT Dataset due to its inconsistency. The "outlier problem" is what these people commonly call this issue. Before the data can be normalized, there is a step that must be taken. Outliers have the potential to cause false positives in the proposed methodology for detecting malicious behaviour. To function, the Median Absolute Deviation Calculator (MADC) needs the formula (Eq. 1):

$$MADC = median(T_{fj} - |median(T_{fj})|) \quad (1)$$

Data normalization process: As part of the normalization procedure, furthermore use the min-max approach to compute the features of the input dataset, here indicated by the T_{fj} integer attribute in the range 0-1 and calculated as (Eq. 2):

$$\tilde{T}_{fj} = \frac{T_{fj} - min(T_f)}{max(T_f) - min(T_f)} \quad (2)$$

One-hot-encoding process: The one-hot-encoding(T_1, T_2, T_3, T_4) technique is necessary to convert them into a numeric value, and it is used for attacks that target the protocol model, the service, or the flag. To illustrate the classification of each feature, a binary number was used. A one-hot-encoding vector was also created from the service and flag features that use T_3 and T_4 symbol representations. The number of attack features in 47 features was represented by 122 dimensions (84 binary and 30 continuous) for each feature.

Features Extraction: The CIC IoT Dataset includes elements that reveal the device's data transfer rates, packet counts, and underlying protocols. Create a new, comprehensive topology with multiple real IoT campaigns serving as attackers or targets, and label them as part of the CICIoT2023 dataset of realistic IoT attacks. In this section, 33 attacks, broken down into 7 categories, are carried out against IoT devices, documented, and collected data from. These features are drawn from current research works for securing the IoT (Dadkhah et al., 2022). Although these features have been employed and verified in previous works, the primary objective is to provide a versatile strategy for training ML models with a large number of features. Therefore, using the scripts developed for this study and the renewed network traffic (i.e.pcaprecords), numerous other aspects can be retrieved or processed.

Furthermore, classify attacks based on the extracted features, ranging from 10 (Backdoor Malware) to 100 (DDoS SlowLoris, DDoS ACK Fragmentations, DDoS PSHACK Floods, DDoS HTTP Floods, DDoS ICMP Fragmentations, DDoS RSTFIN Floods, DDoS ICMP Floods) and ten (Benign Traffics, Host Discovery, Command Injections, Dictionary Brute Force, Operating System Scans, finally used Pandas (Team, 2020; Harris et al., 2020) to merge all the individual files into a single .csv dataset. The generated .csv dataset then reflects the total features of the data.

3) SPC-CNN Model

The SPC-CNN component is responsible for feature extraction within the SCEEHO-SPC-CNN-CD-DBN model. The SPC enhances traditional CNN architectures by breaking down the convolutional process into smaller, more focused operations. This division allows the network to capture more diverse

and fine-grained features, which is particularly useful in identifying the slight variations associated with different types of cyber-attacks. By employing SPC, CNN becomes more robust in handling data variations, making the feature extraction process more effective and efficient.

Initially, in SPC-CNN, the original k×k matrix undergoes convolution with Conv1, resulting in the generation of L feature maps. There exist several resemblances among these feature maps. Nevertheless, these patterns exhibit slight variations, proposing the presence of a certain level of redundancy in inter-channel information. Nevertheless, the conventional attack recognition approach that relies on Convolutional Neural Networks (CNN) (Jo et al., 2020) fails to consider the redundancy present in channels. This model simply performs direct convolutions on all feature maps in the subsequent convolutional layer, without conducting any feature analysis. The presence of redundancy on these channels is likely to impact the network's detection performance and execution efficiency. Nevertheless, the elimination of these characteristics is not an easy task due to the inherent challenge of discerning if these identical pattern aspects encompass distinct particulars.To effectively extract multiscale features and minimize inter-channel information redundancy more rationally, furthermore propose the utilization of an SPC-CNN architecture that incorporates the SPCConv module. As depicted in Figure 3, the SPC-CNN architecture comprises seven distinct layers. These levels include an input layer, one convolution layer, two SPCConv modules, a fully connected (FC) layer that incorporates the CD-DBN, a softmax layer, and an output layer. The SPCConv module comprises four distinct components, namely a channel splitting block, two convolution blocks, and a feature fusion block. The following is the fundamental SPC-CNN process.

- Conv1 produces ℒ feature maps in Part 1, in which the channel splitting module divides into representative and redundant sections. The SPCConv module divides L input channels into a representative part and a redundant part based on α: $(1-\alpha)$.
- The SPCConv module's Parts 2 and 3 use various amounts of feature extraction for the redundant and representative parts, respectively.
- In the fourth section, a soft attention module is utilized to integrate the features from various channels, which can help make the application of those characteristics more logical.
- To get the model to converge, furthermore determine the loss value based on the Softmax layer's output and then improve the network parameters using error backpropagation.

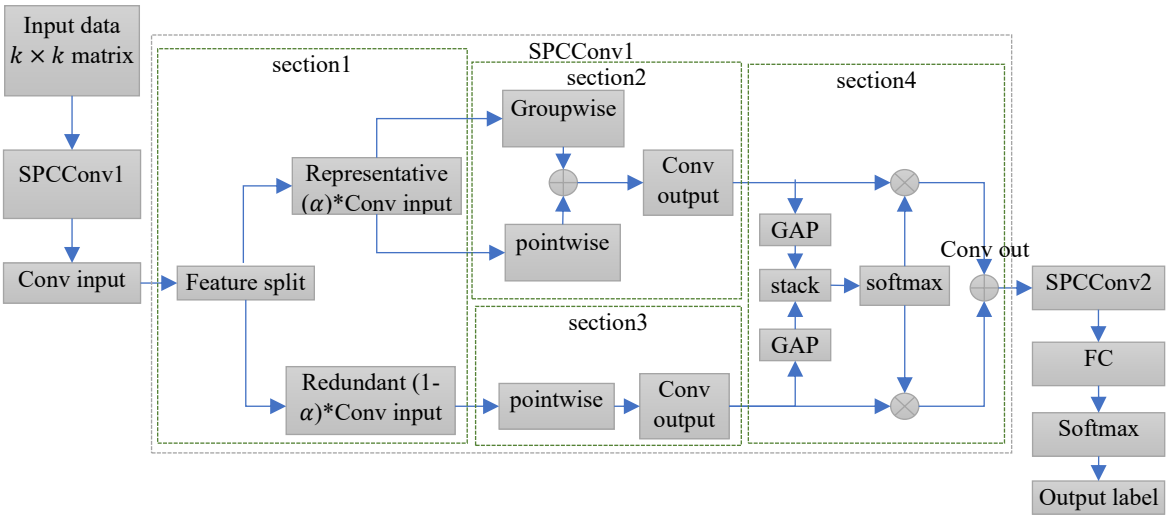

Figure 3: Block Diagram of SPC-CNN Model

4) CD-DBN Model

The DBN component, integrated with CD, focuses on feature learning and classification. DBN is a generative graphical model that consists of multiple layers of stochastic, latent variables, making it adept at learning deep representations of features extracted by CNN. The CD algorithm accelerates the training process by approximating the gradient of the log-likelihood, improving the DBN's ability to distinguish between normal and malicious behaviour. This capability is particularly valuable in dealing with high-dimensional and non-linear data, which are common in cyber-attack scenarios.

The DBN has quatern layers of pre-trained RBM and an additional layer on top for the output (Softmax Regression) which is combined in the FC layer of SPC-CNN. To represent and classify attacks using DBN (Balakrishnan et al., 2021), parameters must be anticipated through model training. Pre-training for presentation and fine-tuning for classification makeup CD-DBN training. At the same time, the result of the CD-DBN calculation that was performed on the model was sent to the input of the Softmax Regression procedure. The RBM stacks that make up the CD-DBN have been expanded to include Softmax Regression.

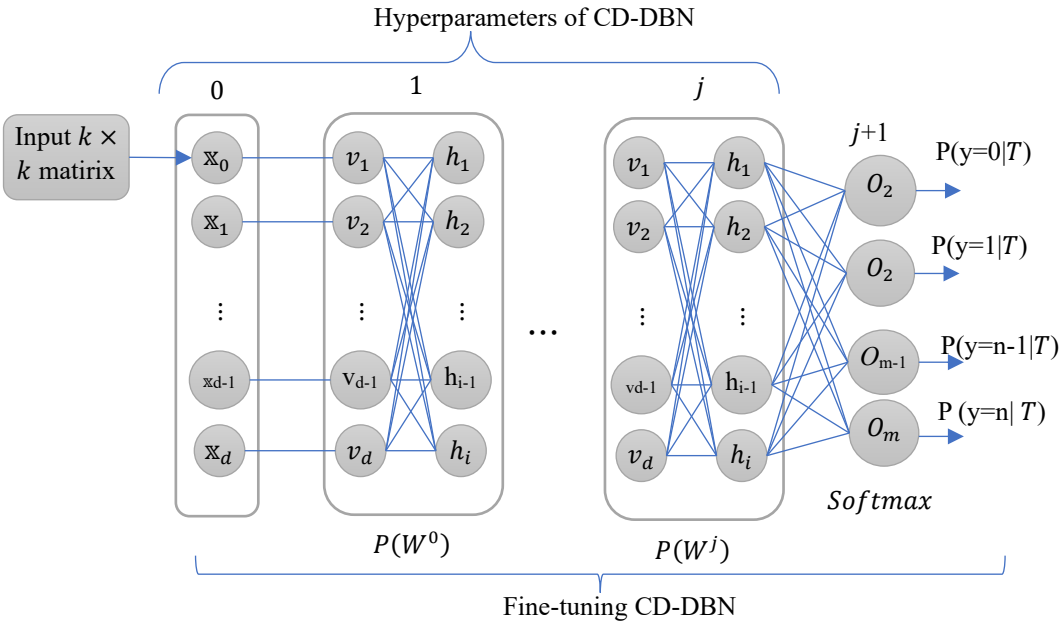

Figure 4: Architecture of CD-DBN

During the initial stages, the CD-DBN undergoes training to reconfigure training material that lacks tags, thereby rendering the initial stages as unsupervised in Figure 4. In the proposed model, Exp_{data} and Exp_{model} Are expectations of probability. The CD-DBN is unsupervisedly trained during the preliminary stages to rearrange untagged training data. Both Exp_{data} and Exp_{model} are probability expectations in the proposed model (Eq. 3, 4 & 5).

$$\frac{\partial \log Pr(v,h)}{\partial W_{ij}} = Exp_{data}[h_j v_i] - Exp_{model}[h_j v_i] \quad (3)$$

$$\frac{\partial \log Pr(v)}{\partial a_i} = Exp_{data}[v_i] - Exp_{model}[v_i] \quad (4)$$

$$\frac{\partial \log Pr(v)}{\partial b_j} = Exp_{data}[h_j] - Exp_{mod\ el}[h_j] \quad (5)$$

All three of the original methods used in conventional DBN networks (Eqs. 8, 9, and 10) failed to produce the second term directly in the proposed model. Because the DBN system can learn the probability embedded in the distribution. This probability can be computed using Gibb's sampling. However, this strategy is too time-consuming and impractical to use in real life. The model employed the fast-learning contrastive divergence (CD) algorithm (Hinton, 2002) to obtain a suitable answer to this challenge. The initial Markov chain was terminated with a training example. After *k* iterations of Gibb's sampling, samples were collected and labelled CD-k. After reviewing the research, furthermore find that the CD will perform adequately even at k=1. To train stacked RBMs layer by layer to produce the DBN network, the proposed model does the following updates to the parameter W following CD-1 (Eq. 6).

$$W^{ts+1} = W^{ts} + \left(Pr\left(\frac{h}{v^{(0)}}\right)\left[v^{(0)}\right]^T - Pr\left(\frac{h}{v^{(1)}}\right)\left[v^{(1)}\right]^T\right) \quad (6)$$

Where is the Ɛ training frequency in ts Time steps. Model hidden variables are denoted by $h = \{h_m\}$ And model visible variables by $v = \{v_m\}$. Together, *v* and *h* form a probability distribution that is shown as (Eq. 7).

$$\Pr(v,h) = \frac{1}{PF}\mathrm{e}\big(-Exp(v,h)\big); PF = \sum_v \sum_h \big(-Exp(v,h)\big) \quad (7)$$

The partition function (PF) distinguishes between M features in the outer layer and N attributes in the inner layer. In the CD approach, the w of each feature vector is initially sampled at random inside the network. Following are the directives for putting into action the greedy layer-wise training method across all DBN layers. Initial RBM training uses, as input *x*, data fitting the W_1 Parameters. During RBM and subsequent binary feature layer training, W_1 Is held constant and the trained RBM's h_1 is used as follows:$Q\left(\frac{h_1}{v}\right) = Pr\left(\frac{h_2}{h_1}, W_2\right)$. Freezing W_2, which defines two-layer features, and allows to use h_2 to retrieve the information necessary to train binary features at the third layer: $\left(\frac{h_2}{h_1}\right) = Pr\left(\frac{h_2}{h_1}, W_2\right)$.This procedure is repeated infinitely deep into the underlying layers. Classic binary classification makes use of logistic regression. However, due to the DBN network's support for multiple classifications, Softmax Regression is favoured. Where *m* is the total number of training set instances, and $T^i_{(1)}, y_{(1)}$ is the hidden vector of the most advanced RBM, furthermore write the training set asset $\left\{\left(T^i_{(1)}, y_{(1)}\right), \left(T^i_{(2)}, y_{(2)}\right), \ldots., T^i_{(m)}, y_{(m)}\right\}$. For each class *i*, the conditional probability of $Pr(\, y = j|T^{(i)})$is determined as follows using the Softmax function in the output layer, where $j = 0, \ldots, k$.

$$Pr\big(\, y = j\big|T^{(i)}\big) = \phi_{soft\ max}\big(T^{(i)}\big) = \frac{e^{T^{(i)}}}{\sum_{j=0}^{k} e^{T^{(i)}_k}} \quad (8)$$

where $T_{(i)} \in R^{n+1}$ represents the privileged vector. The definition of *T* is specified in Eq. 9

$$T = w_0 \mathbb{x}_0 + w_1 \mathbb{x}_1 + \cdots + w_m \mathbb{x}_m = \sum_{l=0}^{m} w_1 \mathbb{x}_1 = w^T \mathbb{x} \quad (9)$$

Input: Processed Dataset from SPC-CNN
Output: Trained network for attack detection
1. Train the initial layer as an RBM with the input *h* (0) as the exposed layer.
2. As shown in Eqs. (3 & 4), the output of the first layer can be used as input to the second layer.
3. Train an RBM in the second layer
4. Do steps 2 and 3 until reaching the desired number of layers.
5. Random Weight initialized as in Eq. (6)

6. Update the weight of Edge based on probability using Eq.(7)
7. The individual activation probabilities for visible layer areas in Eq. (8)
8. The Softmax function in the output layer as in Eq. (9)
9. Return the class label generation results.

The pseudocode outlines the training process of a DBN for cyber-attack detection by utilizing the processed data set obtained from the SPC-CNN model. Initially, the network begins by training the first layer as a RBM, where the input layer serves as the visible layer. The output from the first RBM layer was then fed as input to subsequent layers, repeating the process of RBM training until the required number of layers was achieved. This stepwise pre-training helps the network to learn hierarchical feature representations effectively. Random weights are initialized, and these weights are updated iteratively based on probabilistic calculations, enhancing the network's learning capabilities. The individual activation probabilities of visible units are calculated, guiding the network towards more accurate predictions. The output layer employed a Softmax function to generate the final class labels, which indicate the presence of cyber-attacks. The overall approach ensures that the network is well-trained to classify attack types by progressively refining feature representations and optimizing weight adjustments throughout the layers.

4 Hyperparameter Tuning of SPC-CNN and CD-DBN Model Using SCEEHO

The SCEEHO algorithm plays a crucial optimization role within the proposed model. It is inspired by the social behaviours observed in sea crows and elephant herds and is used to fine-tune the hyperparameters of the CNN and DBN components. By optimizing learning rates, weights, and biases, SCEEHO enhances the model's ability to learn complex data patterns, which is critical for accurate cyber-attack detection. This optimization addresses challenges like overfitting, convergence speed, and the generalization capabilities of the model, ensuring that the CNN and DBN operate under optimal conditions.

The integration of SPC-CNN and DBN, optimized through SCEEHO, creates a hybrid model that utilizes the advantages of each component. The CNN efficiently extracts relevant features, while the DBN, enhanced by CD, excels in classifying these features. SCEEHO ensures that both models are finely tuned to work together seamlessly, leading to a system that maximizes detection performance. This hybrid approach effectively addresses the limitations seen in earlier works, such as high computational costs and limited detection capabilities, by creating a model that is both highly accurate and adaptable to different types of cyber-attacks, including those with subtle or evolving signatures.

Both the CD-DBN and SPC-CNN weights are tweaked to optimal values using the adopted SCEEHO. In Figure 5, furthermore,see the input solution after the SCEEHO scheme has been applied to it; the weights of the CNN ($W_1, W_2, \ldots . W_N$), the CD-DBN ($w_1, w_2, \ldots . w_{mn}$), the SPC-CNN (N), and the CD-DBN (mn) are all shown. Eq. (10) lays out the fitness objective of the chosen detection model. Here, Loss represents the errors in detection.

$$Obj = \min(Loss) \tag{10}$$

The attack detection error by a feature vector formulated like equation (11):

$$Loss = \frac{1}{n}\left(\sum_{i=1}^{n}(\bar{O}_i - O_i)^2\right) \tag{11}$$

Where $\bar{O}_i$ and O_i are the expected and actual numbers of classes of traffic with respect to benign and malicious activity. Each IoT node receives 'n' random samples of network traffic.

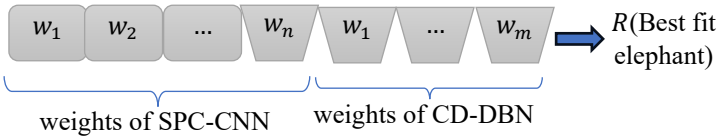

Figure 5: Structure of Solution Encoding for Hypermeter Values Selection

Combining the principles of EHO (Wang et al., 2015) with those of improvement and the Sea Crow Optimization Algorithm (SCOA) (Dhiman & Kumar, 2019), this research develops a novel hybrid SCEEHO system. Despite its superior effectiveness, the conventional EHO lacks the necessary data to properly detect both ongoing and prospective searches. Although 7 constrained real-world industrial applications have been resolved using the current SCOA architecture, the constraints themselves are both time-consuming and computationally complex. This creates a challenge when trying to find optimal solutions. Since the hybrid methods are claimed to be promising for specific search problems with improved convergence speed (Neelakantan & Yadav, 2022), the logic of both algorithms is combined. Here, SCOA logic is combined with EEHO terminology to form SCEEHO. Elephants, being sociable animals, naturally form groupings known as clans, with each clan staying with the Matriarch. Mature male elephants typically spend their time alone. Elephants reproduce at random and quickly form distinct groups based on their relative fitness. The EEHO algorithm relies on the following primary principles.

- There are a certain number of male and female elephants in each elephant clan, and roughly male elephants choose to live apart from the clans to avoid conflict.
- The matriarch (a female elephant) is the undisputed head of all elephant clans and the elephant family.

Clan updating: Each clan in this operator is updated independently. The clan updating matriarchs traditionally have a say in where the herd goes next, as do all of the elephants in clan h. As shown in Eq. (12), however, the proposedSCEEHO approach employs the SCOA updation role for clan updation.

$$R_b(\tilde{it}) = \hat{B} \times R_{best}(\tilde{it}) - R_{\hat{i}}(\tilde{it}) + Levy(\zeta) \quad (12)$$

In Eq. (12), R_b denotes the positions of the sea crow search agent $R_{\hat{i}}$ concerning the best-fit search agent R_{best}(i.e., the fittest sea crow), $R\tilde{\imath}$denotes the search agent's present position, it refers to the present iteration, and the behaviour $\hat{B}$is randomized and used for appropriate comparison between exploration and exploitation. Eq. (13) also provides the update for the optimally sized elephant.

$$R_{n,h,k} = \eta \times R_{cen,h} \quad (13)$$

In Eq. (13), $\eta[0,1]$ Is the midpoint of clan h. The new distinct$R_{n,h,k}$is stated from the data gained by all elephants in clan$h$. $R_{cen,h}$signify the centre of clan h, and it is specified in Eq. (14).

$$R_{cen,h} = \frac{1}{Gh} \times \sum_k^{Gh} R_{h,k,d} \quad (14)$$

In Eq. (14), $1 \leq D \leq \tilde{d}$signifies the Dmeasurement and $\tilde{d}$ indicates the total sizes. Gh specify the number of elephants in clan$h$.$R_{h,k,\tilde{d}}$refers to the $\tilde{d}$ dimensions of the elephant'sdistinct$R_{h,k}$.

Separating operator: In clan starts, adult male elephants are segregated from the females. The operator used for separation is calculated as a byproduct of optimizing the issue. The worst fitness of an elephant at

each group in the separation operator is specified according to the presented evaluation given in Eq. (15), which is part of the SCEEHO logic presented for improving searchability.

$$R_{worst,h} = R_{min}.rand(R_{best} - R_{worst}) * c_i \quad (15)$$

Here, R_{min}denotes the smallest allowed gap between elephant locations. R_{worst} denotes the worst elephants in the clan c_i, and the rand value is derived from the Chebyshev chaotic map. A value between 0 and 1 is possible. The chosen method's pseudo code is described in Algorithm 1 (Eq. 16).

$$R_{q+1} = \cos\left(\tilde{q}\cos^{-1}\left(R_{\tilde{q}}\right)\right) \quad (16)$$

Algorithm 1: Pseudocode of SCEEHOMethod for Hyperparameter Optimization

Beginning of elephant swarm, time stamp, number of clans and fixed number of females.
1. Initialization of weights of SPC-CNN and CD-DBN as elephants
2. Compute the elephant fitness
3. **Repeat**
4. Assemble all the elephants through the calculationof fitness
5. **Clan updating**
6. For h = 1 to$n.\,c_i$ do
7. For k = 1 to $n.\,c_i$do
8. If $R_{h,k} = R_{best,c_i}$then
9. Clan updating by the elephant location using Eq. (13).
10. Else
11. Projected clan updating by the sea crow position using Eq. (12)
12. End if
13. End for k
14. End h
15. **Separating operator**
16. For h = 1 to $n$clan do
17. For each iteration do
18. Substitute the worst elephant as per the proposed Eq. (15)
19. End for h
20. Estimate population by the new updated positions of elephants
21. Assume new update positions of elephants as hyperparameter values
22. **Until the max iteration reached**
23. Return the updated elephant’s position as hyperparameter values.

The pseudocode describes the SCEEHO method for hyperparameter optimization of the SPC-CNN and CD-DBN models. The process begins with initializing the weights of these models as elephants within an elephant swarm, including parameters like time stamps, the number of clans, and the number of females. The fitness of each elephant is calculated, and the elephants are assembled based on their fitness values. The optimization proceeds by updating each clan's positions; if an elephant is the best in its clan, its location is updated using a specific equation (Eq. 13), otherwise, it is updated using the sea crow's position (Eq. 12). A separating operator identifies the worst-performing elephants, which are replaced according to a proposed equation (Eq. 15). The positions of the elephants are then updated, and these positions correspond to the hyperparameter values. This process repeats until the maximum number of iterations is reached, and the final positions of the elephants are returned as the optimized hyperparameter values.

5 Experimental Results and Analysis

1) Experimental Setup

The performance evaluation of the proposed research model is conducted through experimental computation in this section. The experiments were conducted using Python 3.6.0, TensorFlow, and Keras APIs. The research model was executed with Keras APIs and operates on a Google Colab platform equipped with an NVIDIA GTX 1050 4GB GPU. The experimental setup includes a system with 12 gigabytes of RAM, a 1 terabyte hard disk drive, and a 256-gigabyte solid-state drive. This extensive analysis focuses on the effectiveness of the proposed research model, considering many metrics like precision, accuracy, throughput, recall, and f-measure.

In this procedure, an IoT architecture consisting of 105 nodes is subjected to 33 separate attacks. Malicious IoT devices carry out these attacks, which can be categorized into seven types (DoS, DDoS, Recon, spoofing, Web-based, brute force, and Mirai). This dataset allows IoT experts to work with data in a variety of formats to create novel security analytics solutions, and it contains many attacks that are unavailable in other IoT datasets. By using a comprehensive topology with a wide range of IoT devices, executing several attacks that have never been seen in a single IoT security dataset, and analyzing the efficacy of commonly used DL methods in various classification scenarios, the CICIoT2023 dataset expands existing IoT security insights beyond those found in state-of-the-art approaches. Furthermore, these metrics were utilized to analyze how well the proposed SCEEHO-SPC-CNN-CD-DBN model performed. Some of the most common quality indicators used are as follows:

Recall: Recall measures the number of positive class calculations made out of all positive instances in the dataset as specified below (Eq. 17).

$$Recall = \frac{TP}{TP+FN} \tag{17}$$

Precision: Precision measures the number of positive class calculations that belong to the positive class and it is projected as follows (Eq. 18).

$$Precision = \frac{TP}{TP+FP} \tag{18}$$

F-measure: F-Measure carries a single score that balances both the concerns of precision and recall in one number and it is assessed as follows (Eq. 19):

$$F - Measure = \frac{(2 * Precision * Recall)}{(Precision + Recall)} \tag{19}$$

Accuracy: The proportion of successfully segmented data relative to the total number of samples is a popular metric for evaluating classification performance (Eq. 20).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{20}$$

In this scenario, True Positive (TP) means that an attack has been accurately discovered. When an action is appropriately identified as normal, it is said to be true negative (TN). When an action is falsely labelled as malicious, this is called a false positive. When an attack is not detected and falsely classified, this is known as a false negative (FN).

Throughput is the rate at which output is generated and evaluated based on the quantity of flow (Alsamiri, & Alsubhi, 2019).

Evaluation of LSTM, DCNN, and ENN performance as binary classifiers. Additionally, the three DL models were independently deployed and evaluated for evaluation of the binary classifier's performance on the testing dataset. In addition,the optimal binary classification model in terms of evaluation parameters are determined by evaluating accuracy and loss during model training and validation. Table 2 summarizes the outcomes of the binary classification model using the developedSCEEHO-SPC-CNN-CD-DBN, ENN, DCNN, and LSTM-based classifiers. SCEEHO-SPC-CNN-CD-DBN outperforms other LSTM, ENN, and CNN-based classifiers in terms of accuracy and precision. The batch-trained binary classification model relied solely on the SCEEHO-SPC-CNN-CD-DBN. The numerical results are given in Table 3.

Table 3: Binary Classification Evaluation Results Comparison

Model	Traffic Type	Accuracy	Precision	Recall	F1-Score	Throughput (Ms)
SCEEHO-SPC-CNN-CD-DBN	Normal	98.95	98.56	98.95	97.56	0.07
	Attack		99.42	98.32	96.5	
LSTM	Normal	98.2	93.6	97.3	96.3	0.065
	Attack		96.7	94.2	94.3	
DCNN	Normal	94.58	92.3	89.33	85.6	0.04
	Attack		93.4	95.7	90.43	
ENN	Normal	93.54	90.6	87.2	82.5	0.03
	Attack		91.3	90.3	86.7	

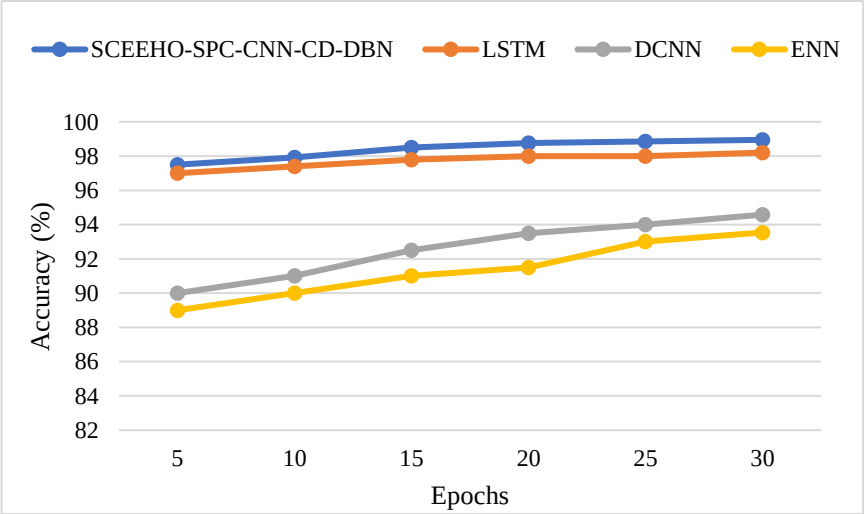

Figure 6: Accuracy Results Comparison

For a given epoch count in a given database, Figure 6 shows the degree to which the developedSCEEHO-SPC-CNN-CD-DBN agrees with the currently employed models. Processing times are reduced by the SCEEHO-SPC-CNN-CD-DBN without any loss of accuracy. The SCEEHO-SPC-CNN-CD-DBN outperforms all other classification models, including the LSTM, DCNN, and ENN, with an accuracy of 98.95% without requiring a significant number of epochs during reduction. The test findings also show that the SCEEHO-SPC-CNN-CD-DBN is superior to the individual ENN, DCNN, and LSTM models. The developed model shows superior feature extraction and outperforms state-of-the-art approaches by a wide margin.

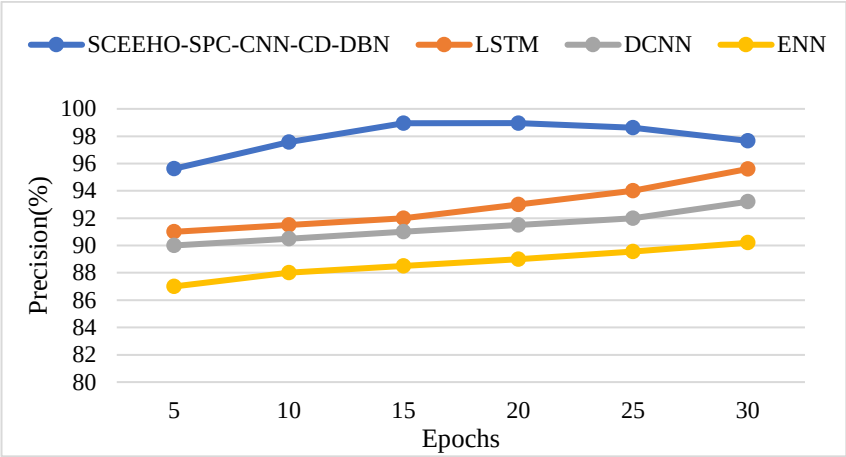

Figure 7: Precision Results Comparison

For a given subset of database properties, Figure 7 shows how the proposed SCEEHO-SPC-CNN-CD-DBN compares to other models such as LSTM, DCNN, and ENN in terms of accuracy. The precision improves with time as measured in epochs. For example, the SCEEHO-SPC-CNN-CD-DBN has a higher accuracy than earlier approaches (97.65%). When SPC-CNN and CD-DBN are combined, they produce the greatest results in terms of loss values and overall performance across precision metrics. Experiments show that the fusion-based hybrid can extract better features for better classification. Using the SCEEHO method yields better true negative and true positive rates than conventional approaches. To boost the model's performance and drastically cut down on false negatives, a hybrid feature selection technique is fed into it. That the proposed SCEEHO-SPC-CNN-CD-DBN technique outperforms the standard CNN and DBN model demonstrates its importance.

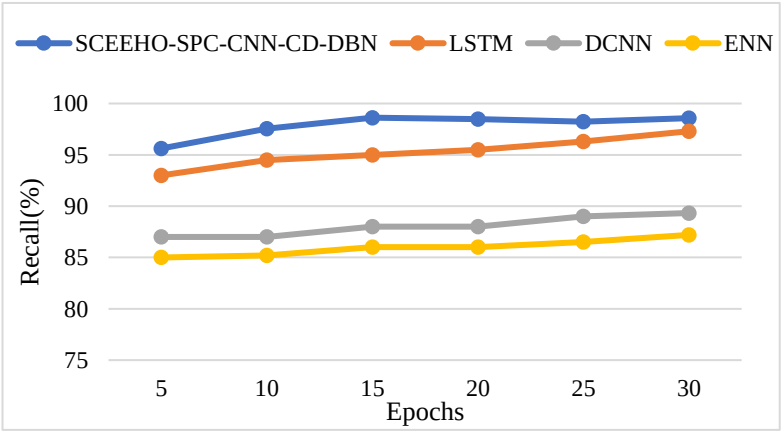

Figure 8: Recall Results Comparison

Traditional neural network models are impacted by the randomness of the initial weight vector and bias coefficient. By providing optimal weight and bias coefficients, the proposed SCEEHO optimization algorithm expedites the convergence of the proposed CD-DBN model. Recall is shown in Figure 8 for the proposed SCEEHO-SPC-CNN-CD-DBN and other current models like LSTM, DCNN, and ENN over a range of feature counts. With more iterations or epochs, the accuracy rises. For example, the SCEEHO-SPC-CNN-CD-DBN achieves a recall of 98.56 percent, which is higher than any of the prior approaches. According to the findings, the developedSCEEHO-SPC-CNN-CD-DBN model has superior cyber-attack classification performance compared to all other models.

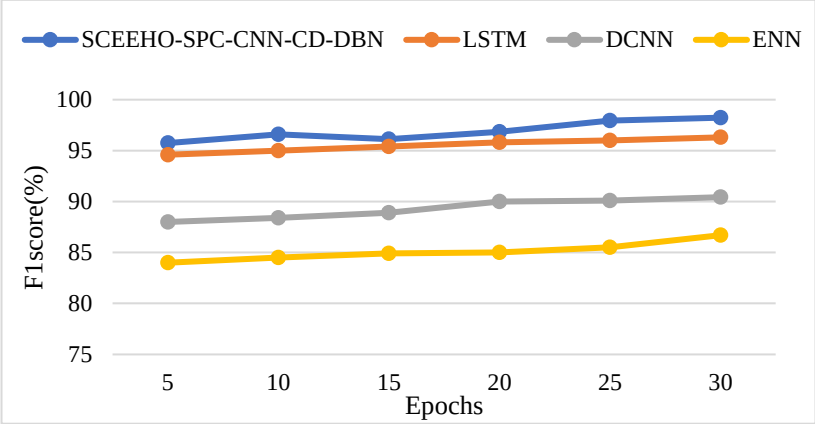

Figure 9: F1-Score Results Comparison

By applying the SCEEHO method independently on the weight and bias vectors, the proposed SPC-CNN and CD-DBN model is fine-tuned. The proposed SCEEHO-SPC-CNN-CD-DBN has a higher F1-sore for the number of features in the provided databases than the existing SCEEHO-SPC-CNN-CD-DBN, LSTM, DCNN, and ENN models (see Figure 9). The epoch count and the f-measure are both tuned at the same time. When compared to other models, the SCEEHO-SPC-CNN-CD-DBN achieves an f-measure of 98.23%. On both the accuracy and F1 scales, the SCEEHO-SPC-CNN-CD-DBN emerged victorious. These procedures, when applied to SCEEHO, allow for substantial hyperparameter quality improvements for SPC-CNN-CD andDBN at little computational cost, resulting in a high F1 score. The SPC-CNN model is preferred because, despite its reduced complexity, it retains the model's expressive power. More useful information features can be recovered from the dataset using the CD-DBN layer, while the redundant features are eliminated. Additionally, the overfitting issue of the model is considerably reduced when the BN layer is used.

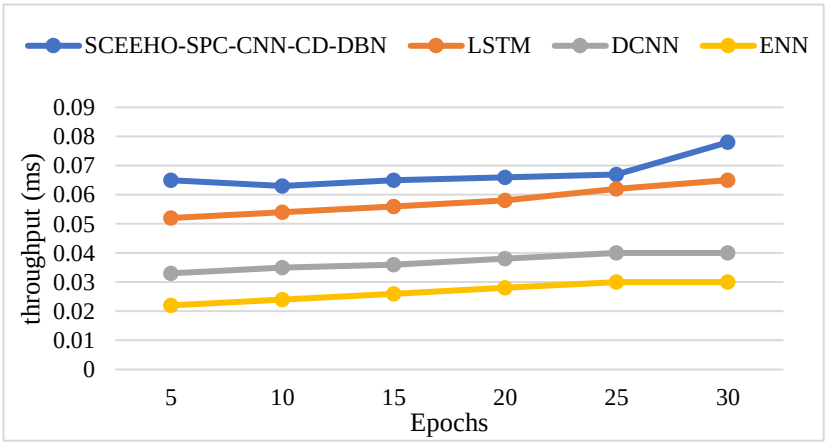

Figure 10: Throughput Results Comparison

Throughput for a variety of feature counts and different models of SPC-CNN-CD-DBN, LSTM, DCNN, and ENN are shown in Figure 10. Throughput is maximized as the number of characteristics increases. A throughput of 0.078ms is achieved by the SCEEHO-SPC-CNN-CD-DBN, which is superior to the other traffic classification techniques. Existing methods are underfitting because they rely on simplistic models that fail to adequately account for the complexities of high-dimensional data. To further increase the proposed method's throughput, it will be utilized to fully train an MCC once the

optimal hyper-parameter configuration has been identified in AHO. A SPC-CNN network analysis model is constructed with the optimized network architecture as its backbone. As a final step in the local feature extraction process, the optimized CD-DBN are layered onto the optimized SPC-CNN model at FC to boost the model's identification and classification performance against network intrusion.

The SCEEHO-SPC-CNN-CD-DBN model achieves the highest accuracy (98.95%) and precision (98.56% for normal and 99.42% for attack traffic), which indicates its ability to correctly identify both normal and attack traffic while minimizing false positives. The recall and F1-score for SCEEHO-SPC-CNN-CD-DBN are also the highest, with recall values of 98.95% for normal traffic and 98.32% for attack traffic. The F1-score further emphasizes the balance between precision and recall, achieving 97.56% and 96.5% for normal and attack traffic, respectively. These metrics demonstrate the model's effectiveness in detecting attack instances without missing any critical cases, thus ensuring reliable attack detection. In terms of throughput, SCEEHO-SPC-CNN-CD-DBN achieves 0.07 ms, which is the highest among the compared models. This reflects the model's efficiency in processing data and making quick predictions, crucial for real-time attack detection scenarios. The technical superiority of the SCEEHO-SPC-CNN-CD-DBN model is due to the hybrid approach combining multiple models (SPC-CNN, CD-DBN) and the optimized hyperparameter selection via SCEEHO.

6 Conclusion and Future Works

This research has contributed to the advancement of security analytics applications by introducing a new and comprehensive IoT attack dataset for use in actual IoT operations. The raw data was collected and applied for preprocessing and standardization. The features were extractedfrom the preprocessed and standardized data. An improved hybrid DL model was then fed the derived features to identify attacks. Models like CD-DBN and SPC-CNN are combined in the proposed hybrid classifier. An algorithm called SCEEHO was developed for optimizing the ideal weights, which would allow for more precise and accurate detection. EEHO and SCOA are combined together in SCEEHO. In cases where EEHO is unable to determine the optimal location for a clan, SCOA is employed. The performance of the research model includes the accuracy (98.95%), precision (97.56%), recall (98.56%), f1-score (98.23%), and throughput (0.078ms) and these results of the research model are comparedwith the existing approaches. In future, a novel ensemble DL classification algorithm can be developed for detecting and labeling threats within IoT network traffic. Updating the hyperparameters automatically using swarm approaches and automated DL techniques is another important objective.

References

[1] Abirami, A., & Palanikumar, S. (2023). BBBC-DDRL: A hybrid big-bang big-crunch optimization and deliberated deep reinforced learning mechanisms for cyber-attack detection. *Computers and Electrical Engineering*, *109*, 108773. https://doi.org/10.1016/j.compeleceng.2023.108773

[2] Abusnaina, A., Abuhamad, M., Alasmary, H., Anwar, A., Jang, R., Salem, S., ... & Mohaisen, D. (2021). Dl-fhmc: Deep learning-based fine-grained hierarchical learning approach for robust malware classification. *IEEE Transactions on Dependable and Secure Computing*, *19*(5), 3432-3447. https://doi.org/10.1109/TDSC.2021.3097296

[3] Ahmed, M., & Pandey, S. K. (2024). Digital Innovation Management: A Study of How Firms Generate and Implement Digital Ideas. *Global Perspectives in Management, 2*(3), 13-23.

[4] Al-Haija, Q. A., McCury, C. D., & Zen-Sabato, S. (2021). Intelligent self-reliants cyber-attack detections and classifications systems for IoT communications using deep convolution neural

networks. In *Selected Paper from the 12th International Networking Conferences: INC 2020 12* (pp. 100-116).

[5] Alisawi, M., Hammood, L., Ghazi, A., Abdullah, S. S., Al-Dawoodi, A., Ali, A. H., ... & Nawaf, A. Y. (2023, September). Cyber security after COVID 19: A review. In *AIP Conference Proceedings* (Vol. 2839, No. 1). AIP Publishing.

[6] Alnumay, W. S. (2024). Use of machine learning for the detection, identification, and mitigation of cyber-attacks. *International Journal of Communication and Computer Technologies, 12*(1), 38-44. https://doi.org/10.31838/IJCCTS/12.01.05

[7] Alsamiri, J., & Alsubhi, K. (2019). Internet of things cyber attacks detection using machine learning. *International Journal of Advanced Computer Science and Applications*, *10*(12). https://doi.org/10.14569/IJACSA.2019.0101280

[8] Ammi, M., & Jama, Y. M. (2023). Cyber Threat Hunting Case Study using MISP. *Journal of Internet Services and Information Security*, *13*(2), 1-29. https://doi.org/10.58346/JISIS.2023.I2.001

[9] Balakrishnan, N., Rajendran, A., Pelusi, D., & Ponnusamy, V. (2021). Deep Belief Network enhanced intrusion detection system to prevent security breach in the Internet of Things. *Internet of things*, *14*, 100112. https://doi.org/10.1016/j.iot.2019.100112

[10] Bansal, A., & Mahapatra, S. (2017, October). A comparative analysis of machine learning techniques for botnet detection. In *Proceedings of the 10th international conference on security of information and networks* (pp. 91-98). https://doi.org/10.1145/3136825.3136874

[11] Churcher, A., Ullah, R., Ahmad, J., Ur Rehman, S., Masood, F., Gogate, M., ... & Buchanan, W. J. (2021). An experimental analysis of attack classification using machine learning in IoT networks. *Sensors*, *21*(2), 446. https://doi.org/10.3390/s21020446

[12] Dadkhah, S., Mahdikhani, H., Danso, P. K., Zohourian, A., Truong, K. A., & Ghorbani, A. A. (2022, August). Towards the development of a realistic multidimensional IoT profiling dataset. In *2022 19th Annual International Conference on Privacy, Security & Trust (PST)* (pp. 1-11). IEEE. https://doi.org/10.1109/PST55820.2022.9851966

[13] Den, I., Sech, M., Profitieren von, D., Pandemien, P., Präziser als, J., Zuvor, I., & Cyberangriff Gefährdet. (2021). *SonicWall cyber threats report*. SonicWall.

[14] Dey, A. K., Gupta, G. P., & Sahu, S. P. (2023). A metaheuristic-based ensemble feature selection framework for cyber threat detection in IoT-enabled networks. *Decision Analytics Journal*, *7*, 100206. https://doi.org/10.1016/j.dajour.2023.100206

[15] Dhiman, G., & Kumar, V. (2019). Sea crow optimization algorithms: Theory and its application for large-scaled industrial engineering problem. *Knowledge-based system*, *165*, 169-196.

[16] Ding, W., Abdel-Basset, M., & Mohamed, R. (2023). DeepAK-IoT: An effective deep learning model for cyberattack detection in IoT networks. *Information Sciences*, *634*, 157-171. https://doi.org/10.1016/j.ins.2023.03.052

[17] Diwakar., & Roy, J. (2024). The Role of Data Analytics in Digital Transformation: A Study of how Firms Leverage Data for Insights. *Indian Journal of Information Sources and Services*, *14*(4), 29–34. https://doi.org/10.51983/ijiss-2024.14.4.05

[18] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, *29*(7), 1645-1660. https://doi.org/10.1016/j.future.2013.01.010

[19] Gyamfi, N. K., Goranin, N., Čeponis, D., & Čenys, A. (2022). Malware detection using convolutional neural network, a deep learning framework: comparative analysis. *Journal of Interner Services and Information Security, 12*(4), 102-115. https://doi.org/10.58346/jisis.2022.i4.007

[20] Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357-362. https://doi.org/10.1038/s41586-020-2649-2

[21] https://www.unb.ca/cic/datasets/iotdataset-2023.html
[22] Husamuddin, M., & Qayyum, M. (2017, March). Internet of Things: A study on security and privacy threats. In *2017 2nd International Conference on Anti-Cyber Crimes (ICACC)* (pp. 93-97). IEEE. https://doi.org/10.1109/Anti-Cybercrime.2017.7905270
[23] Inayat, U., Zia, M. F., Mahmood, S., Khalid, H. M., & Benbouzid, M. (2022). Learning-based methods for cyber attacks detection in IoT systems: A survey on methods, analysis, and future prospects. *Electronics*, *11*(9), 1502. https://doi.org/10.3390/electronics11091502
[24] Jeong, H. L., Ahn, S. K., Baek, S. H., & Park, K. W. (2019). Anomaly Detection Technology Using Potential Difference Displacement Detection of Data Bus. *Journal of Internet Services and Information Security*, *9*(4), 68-77. https://doi.org/10.22667/JISIS.2019.11.30.068
[25] Jo, W., Kim, S., Lee, C., & Shon, T. (2020). Packet preprocessing in CNN-based network intrusion detection system. *Electronics*, *9*(7), 1151. https://doi.org/10.3390/electronics9071151
[26] Kandhro, I. A., Alanazi, S. M., Ali, F., Kehar, A., Fatima, K., Uddin, M., & Karuppayah, S. (2023). Detection of real-time malicious intrusions and attacks in IoT empowered cybersecurity infrastructures. *IEEE Access*, *11*, 9136-9148. https://doi.org/10.1109/ACCESS.2023.3238664
[27] Khoa, T. V., Hoang, D. T., Trung, N. L., Nguyen, C. T., Quynh, T. T. T., Nguyen, D. N., ... & Dutkiewicz, E. (2022). Deep transfer learning: A novel collaborative learning model for cyberattack detection systems in IoT networks. *IEEE Internet of Things Journal*, *10*(10), 8578-8589. *https://doi.org/10.1109/JIOT.2022.3202029*
[28] Lotfy, B., & Vatankhah, H. (2014). Study of insurance in cyberspace and its infrastructure in Iran and other countries. *International Academic Journal of Science and Engineering*, 1(1), 49–56.
[29] Madakam, S., Lakke, V., Lakke, V., & Luke, V. (2015). IoT: A literature review. *Journal of Computers and Communication*, *3*(05), 164.
[30] Mahmoud, R., Yousuf, T., Aloul, F., & Zualkernan, I. (2015, December). Internet of things (IoT) security: Current status, challenges and prospective measures. In *2015 10th international conference for internet technology and secured transactions (ICITST)* (pp. 336-341). IEEE. https://doi.org/10.1109/ICITST.2015.7412116
[31] Malik, P., Natiyal, L., & Raam, M. (Eds.). (2022). *Machine Learning for Cyber Security* (Vol. 15). Walter de Gruyter GmbH & Co KG.
[32] Manjramkar, M. A., & Jondale, K. C. (2023). Cyber Security Using Machine Learning Technique. In *International Conferences on Application of Machine Intelligences and Data Analytic (ICAMIDA 2022)* (pp. 680-701). https://doi.org/10.2991/978-94-6463-136-4_59
[33] Mehedi, S. T., Anwar, A., Rahman, Z., Ahmed, K., & Islam, R. (2022). Dependable intrusion detection system for IoT: A deep transfer learning based approach. *IEEE Transactions on Industrial Informatics*, *19*(1), 1006-1017. https://doi.org/10.1109/TII.2022.3164770
[34] Muralidharan, J. (2024). Machine learning techniques for anomaly detection in smart IoT sensor networks. *Journal of Wireless Sensor Networks and IoT*, *1*(1), 10-14.
[35] Neelakantan, P., & Yadav, N. S. (2022). Multi objective task scheduling based on hybrid metaheuristic algorithm for cloud environment. *Multiagent and Grid Systems*, *18*(2), 149-169. https://doi.org./10.3233/MGS-220218
[36] Perwej, Y., Haq, K., Parwej, F., Mumdouh, M., & Hassan, M. (2019). The internet of things (IoT) and its application domains. *International Journal of Computer Applications*, *975*(8887), 182. https://doi.org/10.5120/ijca2019918763
[37] Rajkumar, S., Sheeba, S. L., Sivakami, R., Prabu, S., & Selvarani, A. (2023). An IoT-Based Deep Learning Approach for Online Fault Detection Against Cyber-Attacks. *SN Computer Science*, *4*(4), 393. https://doi.org/10.1007/s42979-023-01808-y
[38] Roopak, M., Tian, G. Y., & Chambers, J. (2019, January). Deep learning models for cyber security in IoT networks. In *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)* (pp. 0452-0457). IEEE. https://doi.org/10.1109/CCWC.2019.8666588

[39] Salman, R. H., & Alomari, E. S. (2023). Survey: Homomorphic Encryption-based Deep Learning that Preserves Privacy. *International Academic Journal of Science and Engineering*, *10*(2), 153–163. https://doi.org/10.9756/IAJSE/V10I2/IAJSE1019

[40] Sarhan, M., Layegy, S., Mostafa, N., Galagher, M., & Portman, M. (2022). Features extractions for machines learning-based intrusions detections in IoT network. *Digital Communication and Network*.

[41] Sathish Kumar, T. M. (2024). Developing FPGA-based accelerators for deep learning in reconfigurable computing systems. *SCCTS Transactions on Reconfigurable Computing, 1*(1), 1-5. https://doi.org/10.31838/RCC/01.01.01

[42] Shahin, M., Chen, F. F., Hosseinzadeh, A., Bouzary, H., & Rashidifar, R. (2022). A deep hybrid learning model for detection of cyber attacks in industrial IoT devices. *The International Journal of Advanced Manufacturing Technology*, *123*(5), 1973-1983. https://doi.org/10.1007/s00170-022-10329-6

[43] Shanmugasundaram, V., Srinivasan, G., & Lavanya, M. (2023). Salp swarm algorithm applied to optimal capacitor allocation problem in distribution network for annual cost savings. *International Journal of Applied Science and Engineering*, *20*(3), 1-8. https://doi.org/10.6703/IJASE.202309_20(3).001

[44] Suvarna, N. A., & Bharadwaj, D. (2024). Optimization of System Performance through Ant Colony Optimization: A Novel Task Scheduling and Information Management Strategy for Time-Critical Applications. *Indian Journal of Information Sources and Services*, *14*(2), 167–177. https://doi.org/10.51983/ijiss-2024.14.2.24

[45] Team, T. P. D. (2020). pandas-dev/pandas: Pandas. *Zenodo, February*. https://zenodo.org/record/7979740

[46] Verma, A., & Ranga, V. (2020). Machine learning based intrusion detection systems for IoT applications. *Wireless Personal Communications*, *111*(4), 2287-2310. https://doi.org/10.1007/s11277-019-06986-8

[47] Wang, G. G., Deb, S., & Coelho, L. D. S. (2015, December). Elephant herding optimization. In *2015 3rd international symposium on computational and business intelligence (ISCBI)* (pp. 1-5). IEEE. https://doi.org/10.1109/ISCBI.2015.8

[48] Yogamadhavan, V. K., & Mannayee, G. (2024). An Evaluation of Various Deep Convolutional Networks for the Development of a Vision System for the Classification of Domestic Solid Street Waste. *Journal of Internet Services and Information Security, 14*(2), 268-283. https://doi.org/10.58346/JISIS.2024.I2.017

Authors Biography

Boyella Mala Konda Reddy was born in the year 1987, Andhra Pradesh, India. He received his Master of Computer Application From Sri Venkateswara University, Tirpathi in 2011 Master of Technology in Computer Science & Engineering from JNTU, Ananthapur in 2015, and Master of Business Administration From JNTU, Anantapur in 2018. He is dedicated to teaching field from the last 6 years. His research areas included Data Mining & Advanced Computer Networks & Wireless Sensor Networks & Could Areas. At present he is working as Associate Professor in Akshara Institute of Management and Technology, Tirupathi, and Andhra Pradesh, India.

Dr.A. Abdul Azeez Khan was born in the year 1982, Tamil Nadu, India. He obtained his doctorate degree in the field of computer science in the year 2018. He is having 19+ years of experience put together in industry and academia. At present he is working as Associate Professor in the Department of Computer Applications at B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai-600048. His papers are published in many International Journals & Conferences. His area of interest includes Artificial Intelligence, Machine Learning, IoT, Elearning and Knowledge Management.

Dr.K. Javubar Sathick was born in the year 1984, Tamil Nadu, India. He obtained his doctorate degree in the field of computer science in the year 2018. He is having 15+ years of experience in academics. At present he is working as Associate Professor in the Department of Computer Applications at B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai-600048. His papers are published in many International Journals & Conferences. His area of interest includes Artificial Intelligence, Machine Learning, IoT, E-learning and Knowledge Management.

Dr. L. Arun Raj was born in the year 1985, Tamil Nadu, India. He obtained his doctorate degree in the field of computer science engineering in the year 2017. He is having 15+ years of in academics. At present he is working as Associate Professor in the Department of Computer Science and Engineering at B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai-600048. His papers are published in many International Journals & Conferences. His area of interest includes Artificial Intelligence, Machine Learning, IoT, Networking, Cyber Security.