International Journal of Computer and Engineering Optimization (IJCEO)
Volume 04, Issue 01, January – February (2026)

RESEARCH ARTICLE

SOAL-IOT: SHRIKE BIRD OPTIMIZATION AND CONCRETE AUTOENCODER-BASED DEEP LEARNING FRAMEWORK FOR IOT INTRUSION DETECTION

Sandhya M[1], Aisha Banu W[2 *], Leninisha Shanmugam[3], Arputha Rathina X[4]

[1]Professor, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, 600127 India.
[2]Professor, Department of Computer Science and Engineering, BSA Crescent Institute of Science and Technology, Chennai, 600048 India.
[3]Associate Professor, School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, 600127 India.
[4]Associate Professor, Department of Computer Science and Engineering, BSA Crescent Institute of Science and Technology, Chennai, 600048 India.

[*]*Corresponding e-mail: aisha@crescent.education*

Abstract – The Internet of Things (IoT) consists of interconnected devices that continuously exchange data, making security a critical concern. However, a number of issues pertaining to IoT security and privacy have emerged as a result of its broad use. For IoT-enabled services to be dependable, secure, and profitable, real-time intrusion detection on IoT devices is essential. Current Intrusion Detection Systems (IDSs) frequently face challenges such as a high False Alarm Rate (FAR), mean squared error, and reduced intrusion detection reliability and accuracy. To address intrusion detection challenges in IoT environments, this work proposes the Shrike bird Optimization and concrete Autoencoder-based deep Learning framework for IOT intrusion detection (SOAL-IOT), as illustrated in the diagram. The process begins with IoT network traffic collection, followed by data pre-processing, which integrates data cleaning and normalization to improve high-quality data. Then, Feature Extraction (FE) is performed using a Concrete Autoencoder (CAE) to learn compact and meaningful feature representations. The extracted features are then refined through Shrike Bird Optimization (SBO) for effective feature selection, which reduces redundancy and computational complexity. Finally, the selected features are classified using a Neural Controlled Differential Equation (NCDE)-based model to accurately distinguish between normal and malicious traffic. Experimental findings on the BoT-IoT datasets demonstrate that the proposed model improves overall accuracy by 4.93%, 6.26% and 4.23% over SPOHDL-ID, CST-AFNet, and HIDSIoMT on the BoT-IoT Dataset.

Keywords *– Intrusion Detection, Shrike Bird Optimization, Deep Learning, Neural Controlled Differential Equation, Internet of Things.*

1. INTRODUCTION

The digitization of culture is already being dominated by the IoT using various innovative approaches, including self-driving cars, autonomous machinery, remote medical diagnosis, and others [1]. Moreover, the IoT has also turned into a prosperous showcase because of the introduction of enhanced features that transmit low-cost power-driven gadgets and ubiquitous connectivity [2] [3]. A variety of wireless internet-connected gadgets are mentioned in the Internet of Things [4]. IoT adoption in our community of tech-savvy individuals and cultures is being hindered by security and privacy concerns, but these are outweighed by the advantages of connecting equipment and sensors to convey information on the Internet without human touch [5]. The protections of these networks and systems against potential security threats are being strengthened by continuing legal initiatives [6]. AI approaches, particularly ML and DL, have enhanced IoT network security [7].

Existing studies on DL based intrusion detection for IoT networks, despite achieving high detection accuracy, suffer from several limitations [8]. Real-time performance may be impacted by federated learning-based IDSs' high communication overhead and sensitivity to data heterogeneity across dispersed IoT devices [9]. Dual attention networks, hybrid CNN–LSTM models, and ensemble frameworks are examples of complex architectures that raise computational costs and are therefore less appropriate for deployment on IoT and edge devices with limited resources [10][11]. Methods that rely on a lot of preprocessing, such as converting network traffic into spectrogram images, increase memory overhead and latency

[12]. Furthermore, a lot of models are only tested on benchmark datasets and need big, labeled datasets, which restricts their applicability to a variety of real-world assault scenarios [13]. These studies continue to face similar issues, such as high training time, energy consumption, and limited examination of scalability and real-time implementation [14]. To combat these challenges, a novel SOAL-IOT model has been proposed to accurately detect intrusions and enhance security. The following are the key contributions of this work:

- The key goal of this work is to design an efficient IoT intrusion detection framework that enhance the detection accuracy and reduces the FAR.
- A Concrete Autoencoder-based FE model is used to extract required features from IoT traffic data to enhance the classification performance.
- Shrike Bird Optimization is employed for feature selection to reduce data complexity and enhance computational efficiency.
- The selected features are then classified using a Neural Controlled Differential Equation model to accurately identify normal and malicious activities.
- Precision is used to assess the suggested SOAL-IOT model's performance. F1-score, recall, and accuracy.

The rest of the section is structured as follows: Section 2 presents the literature survey, Section 3 explains the proposed system, Section 4 discusses the results and discussion, and Section 5 concludes the research work.

2. LITERATURE SURVEY

Researchers have developed various approaches for IDS in IoT. The literature review discusses some recent approaches and highlights their limitations.

In 2025 Olanrewaju-George, B. and Pranggono, B., [15] proposed an IDS that promotes IoT network security can be built using artificial intelligence (AI) techniques like ML and DL. Federated Learning (FL), a decentralized method that trains IDS on individual linked devices, can improve data privacy and performance. This study suggests the use of FL-trained supervised and unsupervised DL models to provide IDS for Internet of Things devices.

In 2025 Ishtiaq, W., et al., [16] CST-AFNet is a unique dual attention-based deep learning method for accurate intrusion detection in Internet of Things networks. The model uses BiGRU to capture temporal correlations, multi-scale CNN to extract spatial features, and a dual attention mechanism (channel and temporal attention) to better highlight significant patterns in the data. According to experiments, CST-AFNet performs better and detects more accurately than conventional deep learning models.

In 2025 Imtiaz, N., et al., [17] proposed XIoT, developed a method for explainable IoT threat detection. XIoT uses state-of-the-art deep learning algorithms like CNN to evaluate spectrogram images obtained from IoT network traffic data in order to identify complex and nuanced patterns of attacks. These findings demonstrate how XIoT may improve IoT security by resolving real-world issues and offer IoT networks a robust, scalable, and understandable defense against sophisticated attacks.

In 2025 Zhang, H., [18] presented an examination Deep learning can be used to audit existing intrusion detection systems, find IoT network incursions, and fix IoT security problems. An ensemble learning model was trained using the BoT-IoT dataset, and 21 neural network models were produced. The aforementioned findings demonstrate how deep learning-based methods may enhance IoT network security.

In 2025 Alkhonaini, M.A., et al [19] proposed hybrid deep learning-based intrusion detection (SPOHDL-ID) sandpiper optimizer for a novel BC-assisted IoT platform. An essential part of the SPOHDL-ID design, which offers security, is the IoT platform's intrusion detection and classification process. In the given case, a secure data-sharing process can be carried out with the help of the BC technology.

In 2025 Berguiga, A., et al., [20] presented an IDS (HIDSIoMT) based on DL and designed to be used in IoMT networks. The model suggested will represent a hybrid of the CNN (FE) and the Long Short-Term Memory neural network (LSTM) (sequence data prediction). Use a mog computing structure to deploy the designed IDS on a Raspberry Pi computer. This enables decentralized processing closer to the IoMT devices, increasing responsiveness and minimizing latency.

The literature review indicates that existing models are challenged by such problems as high false rates and low detection accuracy, especially in the case of the identification of new threats. Also, the majority of the techniques lack the flexibility to react to dynamic threats, and they are affected by the unequal training data, thus reducing the overall performance. They are also not good at handling large and complex data. These problems will be covered in the suggested session to discuss the proposed method.

3. PROPOSED METHOD

An IoT intrusion detection system is introduced in this framework. First, information provided by the IoT network is received and sent to be pre-processed, where data cleaning and normalization are done to improve the quality of the data. Next, effective representations are learned through a Concrete Autoencoder (CAE) for FE. The Secretary Bird Optimization algorithm is employed to identify the most informative features. Lastly, the features are inputted into a classifier based on Neural Controlled Differential Equation (NCDE), which successfully classifies the network traffic as normal or malicious.

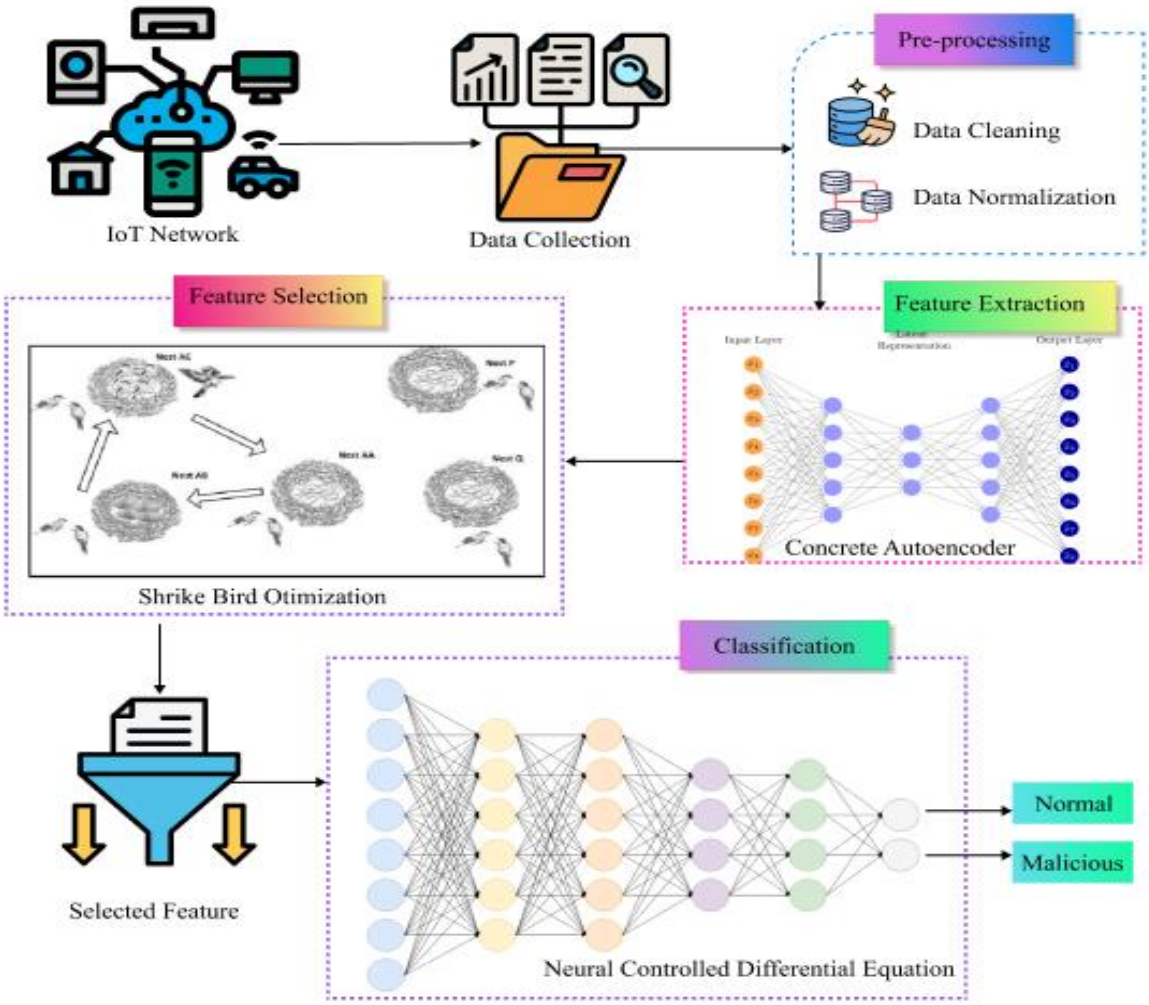

Figure 1. Workflow of SOAL-IOT Method

3.1. Data Collection

This level is where raw data is picked up by various IoT devices that are connected to the IoT network. Some of the information collected is network traffic logs, device communication logs, and sensor-generated data. This data is monitored and stored in a central repository continuously to ensure reliable input to subsequent pre-processing, FE, and intrusion detection processes.

3.2. Data Preprocessing

The preparation of data before to analysis is known as preprocessing. Data cleaning and data normalization are the two preprocessing

Data Cleaning

The step removes wrong entries, redundant records, noise, and missing data in the collected IoT information. The quality of the data is validated through correcting errors and incorporating missing or irrelevant data, and ensuring that the data is correctly and reliably inputted into further processing.

Data Normalization

One of the essential actions in data preparation that is presented to the artificial neural network training is data normalization. It accelerates the attainment of the model and reduces the overall time of training. There are several standardization methods that can be used to standardize data, such as mean normalization, standard scaling, and min-max scaling. Beginning with raw data is a component of data cleaning.

3.3. Feature Extraction using Concrete Autoencoder

The Concrete extraction layer uses Concrete random variables to execute the differentiable FE. In the IoT intrusion detection, the layer can make the model automatically learn the most informative features of high-dimensional network traffic data, and be end-to-end trainable. A concrete random variable is a continuous relaxation of a discrete one-hot extraction vector, which can be optimized using gradients. The degree of relaxation is controlled by a **temperature parameter** $0, S \in (0, \infty)$.

$$n_j = \frac{exp\left((\log \propto_j + h_j)/S\right)}{\sum_{k=1}^{c} exp\left((\log \propto_j + h_j)/S\right)}, j = 1, \cdots, c \quad (1)$$

In this case, the kth element in a certain feature sample vector is denoted by n_j. A discrete distribution is smoothly

approached by the concrete random variable as the temperature parameter $T \to 0$, resulting in one-hot vectors where $n_j = 1$ with probability $\propto_j / \sum_r \propto_r$. The reparametrization approach allows for differentiation with regard to the parameters α, which is an advantage of employing the Concrete random variable in FE. The Concrete extraction layer may learn which features of IoT network data are most instructive for intrusion detection classification in an end-to-end trainable to these capability. Each node in the Concrete extraction layer produces precisely one of the input features when T →0. The output of the jth neuron is equivalent to the following when the network is tested using a discrete arg max layer rather than the concrete extraction layer following training:

$$y_{argmax_i} \propto_i^{(j)}$$

This ensures that the most informative features, as learned during training, are explicitly extracted from IoT network traffic for downstream intrusion detection classification.

3.4. Feature Selection using Shrike Bird Optimization (SBO)

Shrikes belong to the Laniidae family of passerine birds, which are marked by their tendency to impale their flesh on thorns after catching insects, small birds, or mammals.

Shrike Bird Optimization

life cycle of a shrike bird. live in a population that is not in an urban environment; they have a large number of nests, each of which has two parents. The SBO modelled the shrikes' survival and breeding habits. We show the SBO pseudo-code. The pseudo-code explains the SBO's execution procedure in an easy-to-understand manner. Initializing parameters is the first step in the SBO. B is the amount of eggs in each nest that are considered nestlings, N is the total number of nests in the population, and α is a constant that is thought to have an impact on the bird. A population of N nests, each with two randomly generated parent birds, serves as the starting point for SBO's search space. B number of nestlings will be generated after the nests are ready and the population is established.

$$Popolation(M) = \begin{pmatrix} \begin{bmatrix} S_{jn} & S_{je} \\ m_{jk} & m_{jk} \end{bmatrix} & \cdots & \begin{bmatrix} S_{jn} & S_{je} \\ m_{jk} & m_{jk} \end{bmatrix} \\ \vdots & \ddots & \vdots \\ \begin{bmatrix} S_{jn} & S_{je} \\ m_{jk} & m_{jk} \end{bmatrix} & \cdots & \begin{bmatrix} S_{jn} & S_{je} \\ m_{jk} & m_{jk} \end{bmatrix} \end{pmatrix} \quad (2)$$

Each component of a population, such as a pool, represents a nest in which parents are randomly generated and j = (1,2,...,M). There are several solutions in each nest, and the father and nestling are regarded as algorithmic solutions.

$$r_j = AB + rand(CB - AB) \quad (3)$$

In the initialization stage, two birds are created as parents for each nest, and the fittest bird is selected as dominant. The F parent continues to be the female parent, whereas the M parent is the dominating parent. Using equations (3) and (4), each nest produces a B nestling during the breading phase. Nestling is created using 1eggj, where j = (1,2,...B), r is a random value in [-1, 1], and 1eggj is produced from both parents.

$$\Delta egg_j = (Fparent - Mparent) + r \quad (4)$$

$$nestling_j = Fparent + \Delta egg_j \quad (5)$$

Nestlings will look to their parents for help. The nestlings are mostly fed by the dominant male parent, with assistance from the female. In some cases, the male may feed alone, and similarly, the female may feed alone, ensuring that all nestlings receive sufficient nourishment for proper growth and development.

3.5. Classification via NCDE

Assume that we witness a time series $y = (s_0, y_0), \ldots, (s_m, y_m)s$, where $s_0 < \cdots < s_m$ and $s_j \in T$ represents the timestamp of the observation vector $\in (R \cup \{*\})\, v$, where * is used to indicate that some information may be missing. Let $Y_y: [0, m] \to T_u$ be a (continuous, bounded variation) interpolation such that $Y_y(j) = (s_j, y_j)$ where "=" indicates equal value up to missing data. The control path is denoted by Y_y. One of the frequencies of observations may contain information that would be hidden by straightforward interpolation if timestamps are inconsistent or data are unavailable. This is a well-known fact about medical treatment unit statistics. In these situations, we can swap out each $(s_j, y_j)\, 7 \to (s_j, y_j, b_j)$ where $b_j \in T\, u$ counts how many times the channels in xi have been detected up to ti before continuing as before.

Let $g_\theta: \mathbb{R}^v \longrightarrow \mathbb{R}^{v \times w}\ and\ \varsigma_\theta: \mathbb{R}^v \longrightarrow \mathbb{R}^w$neural networks based on learnable parameters θ_1, θ_2. The dimension of the information propagated along the solution trajectory is represented by the value w, a hyperparameter that characterises the size of the concealed state. The Neural CDE model is defined as the solution Y to if Y_y is piecewise continuously differentiable, which will always be the case for us.

$$x(s_0) = \varsigma_\theta(s_0, y_0),\ x(s) = x(s_0) + \int_{s_0}^{s} g_0\big(x(t)\big) \frac{bY_y}{ds} ds\ for\ s \in (s_0, y_0), \quad (6)$$

Therefore, it is possible to understand and solve the model as an ordinary differential equation. In this case, $g_0\big(x(t)\big) \frac{bY_y}{ds}$ represents a matrix-vector product. The response of a CDE driven or controlled by Y_y is referred to as the solution y.

Now functioning in continuous time, the evolving $s_j \in T$ is comparable to the hidden state in an RNN. The model's output is usually a linear map on this hidden state. It can be applied to $x(s)$ for all times $s \in (s_0, y_0)$ to provide a time-dependent output route, or it can be applied to $x(s_m)$ for a single output, like classification.

4. RESULT AND DISCUSSION

The simulations in this part analyze the efficiency of the SOAL-IOT model using measures such as precision-recall

curve, F1 score, recall, precision, and accuracy. The Python 3.6.5 tool is used to duplicate the proposed method on a PC with a 250GB SSD, i5-8600k, GeForce 1050Ti 4GB, 16GB RAM, and 1 TB HDD.

4.1. Performance metrics

Equations (7), (8), (9), and (10) specify standard evaluation metrics that are used to evaluate the suggested Android malware detection. The accurate prediction of malware samples is indicated by True Negative (TN) and True Positive (TP)scores, respectively. The false positive (FP)and false negative (FN) scores demonstrate inaccurate classification, where a typical app is incorrectly tagged.

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (7)$$

$$Pecision = \frac{TP}{(TP+FP)} \quad (8)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (9)$$

$$F1 - Score = \frac{2*Precision*Recall}{Precision+Recall} \quad (10)$$

4.2. Comparison Analysis

The recall, F1-score, accuracy, and precision of the suggested and current methods, including SPOHDL-ID, CST-AFNet, and HIDSIoMT, are compared.

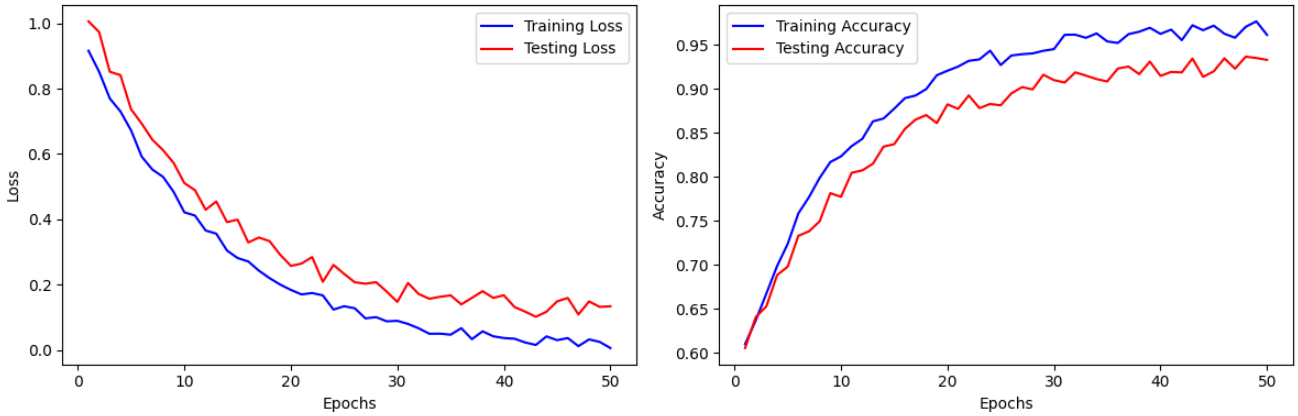

Figure 2. Accuracy and Loss Curve for SOAL-IOT Model

Figure 2 shows the effectiveness of the strategy for identifying the type of intrusion using the BoT-IoT dataset. Figures 2a and 2b display the loss and accuracy curve of the proposed approach. Both the training and testing curves illustrate the learning path of the NCDE approach. The accuracy on the training and testing sets continues to improve as the number of epochs increases, but the loss eventually diminishes. The accuracy rate for classifying invasions using the NCDE approach is 98.3%, with a model loss of 0.07.

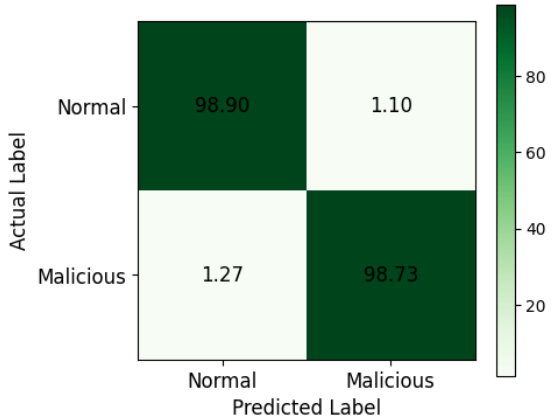

Figure 3. Confusion matrix for SOAL-IOT Model

The SOAL-IOT intrusion detection model's confusion matrix is displayed in Fig. 3. The model correctly classifies 98.90% of normal samples, with 1.10% misclassified as malicious. Similarly, 98.73% of malicious samples are correctly identified, while 1.27% are misclassified as normal. The confusion matrix clearly illustrates the balance between correct and incorrect predictions, highlighting the efficacy of the SOAL-IOT method in distinguishing normal as well as malicious samples.

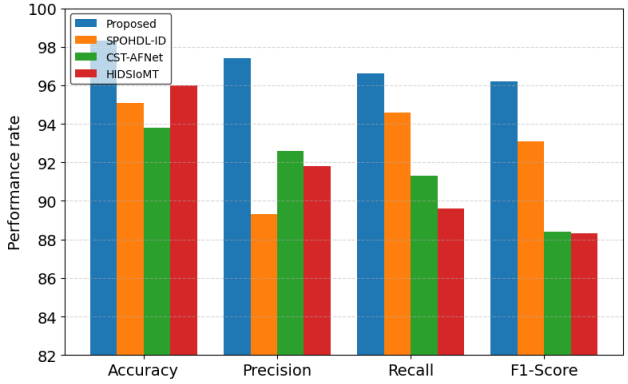

Figure 4. Performance rate comparison

Figure 4 presents the performance rate comparison of SOAL-IOT model with SPOHDL-ID, CST-AFNet, and HIDSIoMT based on F1-score, precision, accuracy, and recall. The findings suggest that the SOAL-IOT model can perform the best in all metrics. It is 98.3% accurate, 97.4% precise, 96.6% recall, and 96.2% F1-score, which has been noted to be better than the current methods. The performance of the other approaches is lower and moderate, which supports the efficiency of the suggested framework to detect the work of the IoT intrusion.

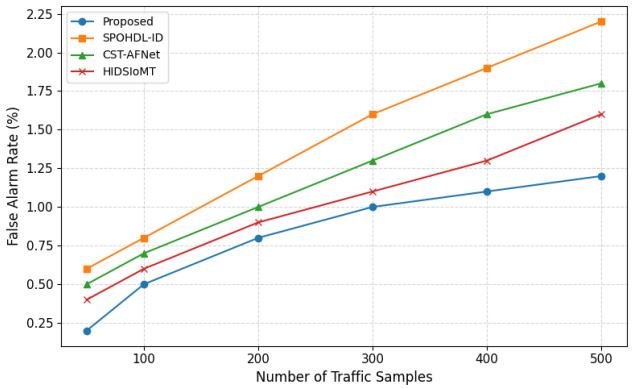

Figure 5. False alarm rate comparison

Figure 5 demonstrates the rate of false alarms of the proposed method in comparison with SPOHDL-ID, CST-AFNet, and HIDSIoMT. FAR grows slightly with an increase in the quantity of samples of traffic of all schemes. Nonetheless, the FAR of the suggested method is always minimal. The SOAL-IOT method has a lower value of 1.2 at 500 samples compared to SPOHDL-ID 2.2, CST-AFNet 1.8, and HIDSIoMT 1.6. This demonstrates the fact that the proposed model generates fewer false alarms.

5. CONCLUSION

In this work, the SOAL-IOT intrusion detection framework with the use of the BoT-IoT data set is introduced. It begins with the collection of IoT network traffic data out of the BoT-IoT dataset, which is followed by data pre-processing to remove noise and normalize data to have homogeneous input. Then, a Concrete Autoencoder with the purpose of FE is used to create small and informative feature representations. The optimized features are further refined by the optimization of features with Shrike Bird Optimization (SBO) to achieve good feature selection, eliminate redundancy, and lessen computational complexity. Lastly, the chosen features are grouped with the help of the model based on Neural Controlled Differential Equation (NCDE) in order to effectively differentiate between normal and malicious traffic. In IoT applications, the FAR is low, and detection performance is enhanced by this hierarchical process. Also, the existing implementation is less flexible to fast-evolving and unknown attack patterns. In order to ensure high reliability of operation in dynamically changing IoT security contexts, future studies will focus on the improvement of computational power, scales, and dynamism.

CONFLICTS OF INTEREST

The authors declare that there is no conflict of interest.

FUNDING STATEMENT

Authors did not receive any funding.

ACKNOWLEDGEMENTS

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

REFERENCES

[1] D. Akgun, S. Hizal and U. Cavusoglu, “A new DDoS attacks intrusion detection model based on deep learning for cybersecurity,” *Computers & Security*, vol. 118, p. 102748, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] K. Coulibaly, “An overview of intrusion detection and prevention systems,” *arXiv*, p. arXiv:2004.08967, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] R. Chataut, A. Phoummalayvane and R. Akl, “Unleashing the power of IoT: A comprehensive review of IoT applications and future prospects,” *Sensors*, vol. 23, no. 16, p. 7194, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[4] K. N. H. Amlan, M. S. Uddin, T. Mahmud and N. B. Riyan, “IoT, cloud computing, and sensing technology for smart cities,” *Intelligent Techniques for Cyber-Physical Systems*, pp. 267–291, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[5] M. Aly and M. H. Behiry, “Enhancing anomaly detection in IoT-driven factories using machine learning approaches,” *Scientific Reports*, vol. 15, no. 1, p. 23694, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[6] N. M. Karie, N. M. Sahri, W. Yang, C. Valli and V. R. Kebande, “A review of security standards and frameworks for IoT-based smart environments,” *IEEE Access*, vol. 9, pp. 121975–121995, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac and P. Faruki, “Network intrusion detection for IoT security based on learning techniques,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[8] M. A. Alsoufi, S. Razak, M. M. Siraj, I. Nafea, F. A. Ghaleb, F. Saeed and M. Nasser, “Anomaly-based intrusion detection systems in IoT using deep learning,” *Applied Sciences*, vol. 11, no. 18, p. 8383, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[9] S. Elsayed, K. Mohamed and M. A. Madkour, “A comparative study of using deep learning algorithms in network intrusion detection,” *IEEE Access*, vol. 12, pp. 58851–58870, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[10] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti and T. H. Kim, “Machine and deep learning solutions for intrusion detection and prevention in IoTs,” *IEEE Access*, vol. 10, pp. 121173–121192, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[11] B. Sun, H. Dai, J. Sun and X. Wei, “Research on Internet of Things intrusion detection model based on graph neural network,” *Engineering Letters*, vol. 33, no. 6, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] A. Basati and M. M. Faghih, “DFE: Efficient IoT network intrusion detection using deep feature extraction,” *Neural Computing and Applications*, vol. 34, no. 18, pp. 15175–15195, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] H. Liu and B. Lang, “Machine learning and deep learning methods for intrusion detection systems,” *Applied Sciences*, vol. 9, no. 20, p. 4396, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[14] R. Abdulhammed, H. Musafer, A. Alessa, M. Faezipour and A. Abuzneid, “Features dimensionality reduction approaches for machine learning based intrusion detection,” *Electronics*,

vol. 8, no. 3, p. 322, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[15] B. Olanrewaju-George and B. Pranggono, "Federated learning-based intrusion detection system for IoT," *Cyber Security and Applications*, vol. 3, p. 100068, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[16] W. Ishtiaq, A. Zannat, A. S. Parvez, M. A. Hossain, M. H. Kanchan and M. M. Tarek, "CST-AFNet: A dual attention-based deep learning framework for intrusion detection," *Array*, p. 100501, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[17] N. Imtiaz, A. Wahid, S. Z. Ul Abideen, M. M. Kamal, N. Sehito, S. Khan, B. S. Virdee, L. Kouhalvandi and M. Alibakhshikenari, "A deep learning-based approach for detection of IoT intrusion attacks," *Photonics*, vol. 12, no. 1, p. 35, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] H. Zhang, "Development of an intelligent intrusion detection system for IoT networks," *Discover Internet of Things*, vol. 5, no. 1, p. 74, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[19] M. A. Alkhonaini, M. A. Alohali, M. Aljebreen, M. M. Eltahir, M. H. Alanazi, A. Yafoz, R. Alsini and A. O. Khadidos, "Hybrid deep learning model for blockchain-assisted intrusion detection," *Alexandria Engineering Journal*, vol. 112, pp. 49–62, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[20] A. Berguiga, A. Harchay and A. Massaoudi, "HIDS-IoMT: A deep learning-based intrusion detection system," *IEEE Access*, 2025. [CrossRef] [Google Scholar] [Publisher Link]

AUTHORS

Sandhya M. is working as a Professor in School of Computer Science & Engineering in Vellore Institute of Technology, Chennai Campus. With over twenty-five years of experience, she has published around 75 papers in reputed International Conferences and journals like IEEE Access, Wireless Personal Communications, Arabian Journal of Science and Engineering, etc. She is a powerful force in the workplace and uses her positive attitude and tireless energy to encourage others to work hard and succeed. Her research interests include Health Care Applications, Wireless Security, RFID, Data Analytics.

Aisha Banu W. is a Professor and Head of the Department of Computer Science and Engineering at B. S. Abdur Rahman Crescent Institute of Science and Technology. She has been serving in the institute since 1997, with over two decades of academic and research experience. She holds a Ph.D. in Computer Science and Engineering from Anna University, Chennai. Her areas of expertise include Cloud Computing, Social Network Analysis, Information Retrieval, and Big Data. She has published more than 68 research papers in reputed journals and international conferences. She has successfully guided 6 Ph.D. scholars in various domains of Computer Science. Dr. Aisha Banu has been involved in multiple funded consultancy and research projects. She has organized and participated in numerous FDPs, workshops, and national conferences. She has served in key administrative roles such as NBA Coordinator, Research Supervisor, and ISO Auditor. She is also a member of the Board of Studies and Academic Council at the institute.

Leninisha Shanmugam M.E,PhD., received the B.E. degree in Computer Science and Engineering from Madurai Kamaraj University. the M.E. degree in Computer Science and Engineering and and Ph.D. from Anna University in Anna University, in Faculty of Information and Communication Engineering. Currently, she is working as a Faculty in School of Computer Science and Engineering, Vellore Institute of Technology (VIT University). Her research interests are feature extraction in remotely sensed images, pattern recognition in Medical Imaging and Agriculture imaging. She has many publications in journals, conferences and book chapters. Especially, more than 10 journals papers in high impact factor journals like Elsevier, IEEE, IET, Hindawi etc. She is reviewer for many reputed journals such as IEEE, IET, Elsevier, etc. She is currently guiding two PhD scholars and one PhD scholar completed her doctorate successfully in last year.

Arputha Rathina X. is currently Associate Professor in the Department of Computer Science and Engineering, B.S.Abdur Rahman University. She has been working in this Institution for the past 25 years. Her area of interest includes Image Processing, Human Computer Interaction and Emotional Intelligence. She has guided more than 40 UG and 25 PG projects. She has visited Hiroshima University - Japan, to acquire the basic knowledge on Kansei Engineering. She did her thesis in the area of Emotion detection using Facial expressions and Speech. She is a member of ACM and IEEE.

Arrived: 07.01.2026
Accepted: 15.02.2026