Cybersecurity in the Internet of Things (IoT): Protecting Connected Devices and Networks

Yan Zhang

School of Information Engineering, Jiangsu Maritime Institute, Nanjing, 211100, China
E-mail: zhyan_1984@163.com

Received 17 February 2025; Accepted 05 December 2025

Abstract

The fast development of the Internet of Things (IoT) has introduced significant threats due to the growing number of connected devices and the vulnerability of data transmitted. Conventional security systems often fail to address the specific requirements of IoT networks because of the limited resources of individual devices and the complexity of the interconnected ecosystem in which they are implemented. The research introduces SecureNet-IoT, a modern security framework that will strengthen the security of connected systems and networks. The framework uses a tuned Intelligent Random Forest (ASSO-IRF) that uses an ASSO to predict the behaviour of IoT devices and distinguish between various types of attacks. SecureNet-IoT actively detects and neutralizes the possible vulnerabilities within the environments of IoT and fog computing. The information was gathered on IoT devices in smart homes and industrial IoTs, including device communication, communication logs, and network traffic, which were pre-processed with a Kalman filter to eliminate noise and normalization techniques to normalize the data. The model classifies devices as authentic, breached, or fake and uses a suggested model to forecast malicious acts. It approximates the possibilities of

Journal of Cyber Security and Mobility, Vol. 15_1, 123–144.
doi: 10.13052/jcsm2245-1439.1515

transitions between states, thus being able to detect threats early. In addition, the framework also analyses device communications to enhance predictive accuracy. The metrics used to assess performance were accuracy (98.96%), F1-score (96.31%), precision (98.78%), and recall (96.24%). The framework demonstrates the system's effectiveness in preventing malicious behavior by successfully categorizing device states, estimating transition probabilities, and analyzing device communications, ultimately enhancing IoT system security and integrity.

Keywords: Cyber security, Internet of Things (IoT), device behavior, adaptive social spider optimizer-tuned intelligent random forest (ASSO-IRF).

1 Introduction

Cyber security (CS) is a group of procedures and technologies which was designed to protect networks, computers, programs, and information from damage, attack, or illegal access. CS is the performance of protecting Information and Communication Technology (ICT) systems from numerous cyber threats or attacks [1]. CS is a crucial ethical relevance since CS technologies have a substantial influence on human well-being as they make feasible modern human organizations that depend on the integrity and accessibility of the data and the computer systems [2].

The primary focus of CS has been protecting individual data from physical and cloud threats. CS threats to digital organizations are an issue for maintaining business growth in the face of evolving technologies for the Social, Mobility, Analytics, and Cloud (SMAC) fields and the IoT, which necessitate the authentication of the novel CS capacities [3]. The IoT concept has offered the world a greater degree of integrity, availability, scalability, interoperability, accessibility, and secrecy in terms of device connectivity [4].

The IoT is defined as an electrical network that links the physical items, including sensors, with the software that enables data collection, analysis, and sharing. The personal healthcare, military, home appliances, and agricultural production structures are a few of the industries that deploy IoT applications [5]. The IoT has certainly transformed everyday life, resulting in a world full of networked gadgets and systems. However, the growth of IoT devices has resulted in a variety of CS risks and vulnerabilities that require immediate attention [6].

A new enhanced security design, SecureNet-IoT, is presented in this research, designed to enhance the security of connected devices and

networks. The system uses the Adaptive Social Spider Optimizer -tuned Intelligent Random Forest (ASSO-IRF) to predict IoT device behaviour and detect different types of attacks.

The study is organized in the following manner: Section 2 is the review of the literature; Section 3 covers the methodology; Section 4 contains the results; Section 5 discusses the advantages of the ASSO-IRF new technique and the limitations of the existing ones; and finally, the conclusion is offered in Section 6.

2 Related Work

With the use of AI models to predict threats, Alterazi et al. [7] enhanced IoT security by the use of Genetic Algorithms (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO) to identify anomalies like U2R and denial-of-service attacks. PSO was also 73 per cent faster than methods currently in use, but its limitations include the possible overfitting and reliance on the quality of datasets, which hinders the generalizability to a variety of IoT conditions. Kalaria et al. [8], using a prey-predator paradigm, demonstrated the use of malicious nodes of IoT networks to deal with the spread of infections using delay differential equations. The approach minimized the infection rates by 90% compared to fuzzy logic and game theory, and minimized the number of infected nodes by 5% and 8% respectively. This is a disadvantage of the model as it is dependent on pre-determined settings. Machine learning (ML) techniques were applied to improve the security of IoT through the analysis of data provided by Strecker et al. [9] to identify the devices and threats. This involved examining the latest developments showing the importance of SDN and fog computing in safeguarding the intense security solutions and the efficiency of the Random Forest (RF) and the K-Nearest Neighbor (KNN) in the anomaly detection method. Even if scaling as well as changing tactics required continuous tweaks to stay effective, results show quite good detection accuracies. To develop CS measures for IoT, Gopalsamy [10] exploited AI-based tools for botnet attack recognition. The Bot-IoT dataset, having about 72 million entries, provides training and testing for ML models such as Decision Tree (DT) and Multi-Layer Perceptron (MLP). Regarding botnet identification, DT has attained high accuracy (99.97%). However, other limitations include possible bias in the dataset and challenges regarding adoption in the real world.

Kalla et al. [11] presented an ML model to classify Portable Executable (PE) files as benign or malicious. Six supervised classifiers were trained

on the static characteristics obtained from 10,868 PE files, with Random Forest giving the best performance of 99%. However, generalizability to other file formats and interaction with commercial antivirus software for practical implementation impose limitations. Predictive Behavioral Mapping (PBM) was used by Blowing et al. [12] to improve ransomware detection. PBM analyses one activity at a time to identify risks before the execution of payloads. PBM yields lower false positives and higher detection rates by using a dynamic scoring scheme and feature integration that is performed recursively. These findings reveal a low computational cost and are very fast in real-time. However, it is associated with such disadvantages as the possible high flexibility of the ransomware to avoid detection and the fact that the greatest accuracy is possible only with large behavioural datasets. Through ML, Umar et al. [13] created a behaviour-based method of detecting malware. This was done with the help of a process of analyzing the malware activity, patterns, and training the model on a sample of 6,999 uncorrupted and 3,540 infected files. High detection rates were realized in the model as it recorded an accuracy of 96.77% with KNN and 98.19% with RF. Limitations include, however, a risk of overfitting and a lesser capability with more sophisticated and dynamic malware encryption schemes. Jamabi et al. [14] improved the prediction of actions and malware detection at early stages using natural language processing (NLP) and ML methods. The framework analysed API call sequences with an API call sequence Model (Bootstrap Aggregating Extreme Gradient Boosting (Bagging-XGBoost)) model, malware was detected by 96.44% and action was predicted by 89.53%. However, this approach uses labeled collections and might experience difficulties with the adaptation to unfamiliar viral types. Based on deep learning (DL) algorithms, Prince et al. [15] explored the IoT security and evaluated risks in smart cities and healthcare in a methodology that encompassed a convolutional neural network (CNN), LSTM-based intrusion detection, and polling of experts. The results show an improved level of security, but there are still problems, such as legislative gaps and a dynamic cyber threat. Rizviet et al. [16] analyzed the vulnerability of IoT devices and have offered security solutions to the household, business, and health care sectors. The research explains how the identification of vulnerability in a systematic manner, analysis of threats, and enforcement of security measures can avert threats, hence the need to implement high levels of security in the IoT environment. Limitations include the lack of standardized regulations and the challenge of mitigating the emerging vulnerabilities in fast-changing technology (Table 1).

Table 1 Overview of the related work

Reference	Methods Used	Dataset	Performance Metrics	Merits	Limitations
Alterazi et al., (2022) [7]	Particle Swarm Optimization for Security	IoT networks datasets	Convergence rate, Security efficiency	Optimized IoT security	High complexity
Saini et al., (2022) [8]	Prey-Predator Model for malicious object detection	IoT datasets	Accuracy, Precision	Novel ecological model for cybersecurity	Model interpretability
Strecker et al., (2021) [9]	ML for IoT Security	ML and IoT datasets	Accuracy, F1-score	ML-driven cyber threat detection	High false positives
Gopalsamy (2020) [10]	AI-based Botnet detection	Botnet traffic datasets	Accuracy	AI-enhanced IoT security	Computational cost
Kalla et al., (2021) [11]	Supervised ML for malware prediction	Malware datasets	Accuracy, Precision, Recall	Predictive malware detection	Limited dataset diversity
Blowing et al., (2024) [12]	Behavioral mapping for ransomware detection	Ransomware datasets	Detection rate	Autonomous threat identification	Complexity in implementation
Umar et al., (2024) [13]	Malware behavioral analysis using ML	Malware datasets	Accuracy, F1-score	Efficient malware detection	High false positive rate
Jamadi & Aghdam (2024) [14]	Bayesian Neural Networks for malware prediction	Malware datasets	Accuracy, Recall	Bayesian uncertainty quantification	Computational intensity
Prince et al., (2024) [15]	DL for IoT security	IoT cybersecurity datasets	Accuracy, Precision	IEEE standard-based security	Limited real-world testing
Rizvi et al., (2020) [16]	Threat modeling for IoT security	IoT network datasets	Threat mitigation rate	Device-level security enhancements	Implementation challenges

Current IoT security systems tackle the issues of high computational complexity, false positives, poor model interpretability, and insufficient real-world testing. In addition, most of the methods have scaling, diversity of datasets, and implementation feasibility problems. To address these inefficiencies, the study presents a best security solution, which is the SecureNet-IoT, that includes the ASSO-IRF. The solution proposed is better in terms of detection, lower false positives, scalability, and real-time cybersecurity in the IoT networks.

3 Methodology

The following steps can be outlined in this section: the first step is to collect data with the help of IoT devices; the second step is to preprocess the data with the Kalman filter (KF) and min-max normalization; the third step is to predict the behaviour of the IoT devices with the help of the ASSO-IRF model and to classify different types of attacks. Figure 1 depicts a representation of the methodological framework.

3.1 Data Collection

The SecureNet-IoT Cyber Behavior dataset was obtained from Kaggle (https://www.kaggle.com/datasets/programmer3/iot-cyber-behavior-dataset/data). The dataset comprises 5000 data samples and provides detailed IoT device activity records, including network traffic patterns, device interactions, and communication logs. Each entry represents a real-time observation from an IoT device, capturing factors such as CPU usage, packet characteristics,

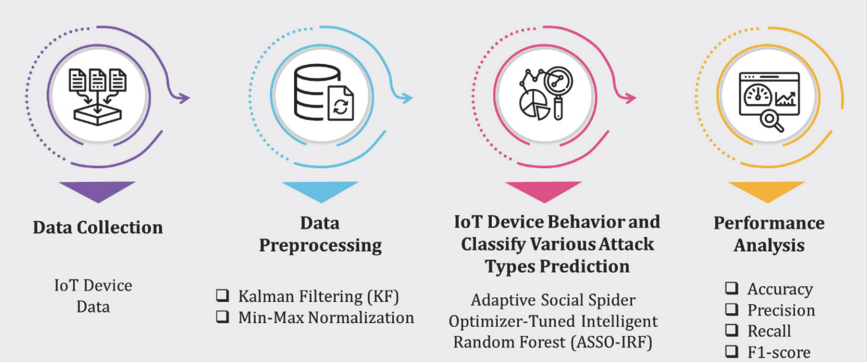

Figure 1 Structure of the methodology.

entropy levels, and authentication attempts. The dataset simulates genuine, compromised, and counterfeit device behavior using realistic IoT threat patterns such as DoS attempts, spoofing, unauthorized port access, botnet activity, and replay attacks. The dataset includes a target label (class_label) with three categories, including Genuine (0): Normal behavior with no threat, Compromised (1): Malicious activity such as DoS botnet, malware, and Counterfeit (2): Device identity, fake firmware, spoofing, abnormal ports. It was divided into 70% for training and 30% for testing to ensure balanced evaluation and reproducibility of the ASSO-IRF model.

3.2 Data Preprocessing

Pre-processing of the data was performed through min-max normalization to standardize the data and a Kalman filter to reduce noise.

3.2.1 Kalman filtering (KF)

The Kalman filter is a sophisticated mathematical model that offers noise-reduction schemes and an optimum prediction approximating a dynamic system. It first filters the IoT data by removing the noise produced by the interactions of the devices and logs of communication and network traffic, which in turn enables accurate detection of anomalies. Improvements in data reliability increase the efficacy of SecureNet-IoT in identifying malicious activities and predicting device behavior. A linear discrete-time dynamic and measurement model, as defined in Equations (1) and (2), is considered.

$$y_l = \Phi_l y_{l-1} + w_{l-1} \tag{1}$$

$$z_l = H_l y_l + v_l \tag{2}$$

Where the state is represented by $y_l \in R^n$, the measurement at time t_l and $\Phi_l \in R^n$ represents the state transition matrix from t_{l-1} to t_l, as depicted by $z_l \in R^n$, $H_l \in R^n$ is the measurement matrix at t_l, Gaussian white noises with zero mean for the process and measurement are denoted by $w_{l-1} \sim N(0, Q_{l-1})$ and $v_l \sim N(0, R_l)$, respectively. The prediction stage of KF is given by Equations (3) and (4).

$$\hat{y}_l^- = \Phi_l y_{l-1} \tag{3}$$

$$P_l^- = \Phi_l P_{l-1}^+ \Phi_l^T + Q_{l-1} \tag{4}$$

Where $\hat{y}_l^-$ and P_l^- are the covariance and predicted state at t_l, respectively. The posteriori covariance at t_{l-1} is given by P_{l-1}^+. The updated stage

is given by Equation (5).

$$\hat{y}_l^+ = \hat{y}_l^- + K_l(z_l - H_l\hat{y}_l^-) \tag{5}$$

Here, the trace of the posterior covariance at time l, as provided by Equation (6), is minimized to get the Kalman gain matrix K_l.

$$K_l = \arg\min T_t(P_l^+) \tag{6}$$

A posteriori covariance in Equation (6) is given in Equation (7), $\hat{e}_l$ is the state estimation error given in Equation (8), and the optimal K_l in Equation (6) is given in Equation (9).

$$P_l^+ = E[\hat{e}_l\hat{e}_l^T] = (J - K_lH_l)P_l^-(J - K_lH_l)^T + K_lR_lK_l^T) \tag{7}$$

$$\hat{e}_l = \hat{y}_l^+ - y_l \tag{8}$$

$$K_l = P_l^-H_l^T(H_lP_l^-H_l^T + H_l)^{-1} \tag{9}$$

3.2.2 Min-Max normalization

Min-max normalization is a data preparation method that rescales structures to a certain range by removing the minimum value and dividing by the data range. It unifies data for the IoT devices, assuring consistency in feeding models such as the ASSO-IRF. This normalization improves model accuracy and performance by reducing the effect of outliers and allows for a more precise prediction of an adverse event. This process is shown in the following Equation (10).

$$Y_{new} = \frac{Y - min(Y)}{max(Y) - min(Y)} \tag{10}$$

Y represents the old value, $max(Y)$ denotes the maximum value, the new value from the normalized results is represented by Y_{new}, and $min(Y)$ denotes the minimum value in the dataset.

3.3 IoT Device Behavior and Classify Various Attack Types Prediction Using Adaptive Social Spider Optimizer-Tuned Intelligent Random Forest (ASSO-IRF)

The ASSO-IRF model combines an IRF with an ASSO model to forecast the behavior of IoT devices and discover concrete forms of cyber-attacks. ASSO adjusts the RF model to enhance the identification and classification process for malicious activities in IoT networks. ASSO got great accuracy in

predictions through fine-tuning of the existing model's parameters, while IRF provides a strong classification scheme capable of differentiating legitimate IoT devices from hacked devices.

3.3.1 Intelligent random forest (IRF)

The IRF algorithm is based on optimization methods to improve the accuracy and effectiveness of the classification. Managing big IoT data thus renders IRF an appropriate tool to shield networks and connected devices from the changing cyber threats. Many missing values often characterize IoT CS datasets; shallow feature spaces make defining meaningful features from noisy ones quite hard. Irrelevant properties in bootstrap samples randomly selected for classification make the classification of malicious activities even trickier. To improve classification accuracy in forecasting IoT device behavior and recognizing cyber risks, the χ^2 test is used as the weighting metric to prioritize important information. Equation (11) is used to calculate a weight for each feature in the feature space, and only a subset of features with high weights is chosen to construct the decision tree.

$$\chi^2 = \sum_{j=1}^{n} \sum_{k-1}^{2} \frac{(O_{jk} - e_{jk})^2}{e_{jk}} \tag{11}$$

Where n is the number of features, the observed value is given by O_{jk}, meaning the count of joint events (A_j, C_k), as shown in Equation (12), and e_{jk} is the expected value of (A_j, C_k), which is defined by Equation (13).

$$O_{jk} = \text{count}(A = a_j \cap C = c_k) \tag{12}$$

$$e_{jk} = \frac{\text{count}(A = a_j) \times \text{count}(C = c_k)}{M} \tag{13}$$

Following the construction of a collection of decision trees, the research combined the output of each classifier using a probability estimation approach. Assume that the test instance is x and each classifier $h_k = (k = 1 \ldots l)$ votes for the potential target class c_j for the input instance x. The output of the classifier is calculated using the formula $P(J(X) = c_j/h_k)$. The ultimate categorization outcomes are then obtained by summing the value of the probability as shown in Equation (14).

$$P(J(x) = c_j) = \frac{1}{l} \sum_{k=1}^{l} P(J(x) = c_j/h_k) \tag{14}$$

The input vector x belongs to class c_j if it has the highest probability. Based on iterative updates determined, the ASSO continues to fine-tune the feature subset selection, as well as the parameter space of the IRF. The global best position, which is represented as Y_g, is a parameter set that maximises the detection accuracy on the validation data. As a result, ASSO acts as an automatic tuner, which automatically sets parameters of IRF, such as feature-weight thresholds, depth of decision trees, and size of ensemble, so that the resulting ASSO-IRF model has an optimal trade-off between exploration (diversity in decision trees) and exploitation (fine-tuning of salient features). This tradeoff produces better attack type classification and prediction of IoT behaviour.

3.3.2 Adaptive social spider optimization (ASSO)

ASSO is a bio-inspired metaheuristic algorithm that mimics the cooperative behavior of social spiders to optimize complex problems. It is used to enhance attack detection by optimizing feature selection and enhancing the classification accuracy of security models. By dynamically altering parameters based on threat patterns, it improves IoT device behavior prediction, efficiently classifies malicious activities, and enhances overall network security. Equations (15) and (16) define the updates of the male and female spiders when the ISSO1 model is updated to include the historical optimum location.

$$F_j^{l+1} = \begin{cases} F_j^l + \alpha \cdot Vib_{d_j} \cdot (Y_d - F_j^l) + \beta \cdot Vib_{c_j} \cdot (Y_c - F_j^l) \\ \quad +\varphi \cdot Vib_{g_j} \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & \delta \leq PF \\ F_j^l - \alpha \cdot Vib_{d_j} \cdot (Y_d - F_j^l) - \beta \cdot Vib_{c_j} \cdot (Y_c - F_j^l) \\ \quad -\varphi \cdot Vib_{g_j} \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & else \end{cases} \tag{15}$$

$$M_j^{l+1} = \begin{cases} M_j^l + \alpha \cdot Vib_{f_j} \cdot (Y_f - M_j^l) + \varphi \cdot Vib_{g_j} \\ \quad \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & \omega > \omega_{N_f+m} \\ M_j^l + \alpha \cdot \left(\sum_{g=1}^{N_m} \left(\omega_{N_f+g} \cdot M_j^l \right) \Big/ \sum_{g=1}^{N_m} \omega_{N_f+g} \right) & else \end{cases} \tag{16}$$

Where Y_g is is the optimal location that the jth spider has discovered. Vibhi is the vibration that the jth spider perceives as a consequence of the effect of the historical ideal location, and φ is a random value between 0 and 1. Early in the search process, the distance is typically great to cause the jth spider to vibrate effectively; therefore, blindfold and random part searches are

used to determine the spiders' motions. Half of the leading female and male spiders are chosen at random to be updated to circumvent the disorganized search for individuals, as shown in Equations (17) and (18).

$$F_j^{l+1} = \begin{cases} F_j^l + \alpha \cdot \chi_1 \cdot (Y_d - F_j^l) + \beta \cdot \chi_2 \cdot (Y_c - F_j^l) \\ \quad +\varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & \delta \leq PF \\ F_j^l - \alpha \cdot \chi_1 \cdot (Y_d - F_j^l) - \beta \cdot \chi_2 \cdot (Y_c - F_j^l) \\ \quad -\varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & else \end{cases} \tag{17}$$

$$M_j^{l+1} = \begin{cases} M_j^l + \alpha \cdot \chi_4 \cdot (Y_f - M_j^l) + \varphi \cdot \chi_5 \\ \quad \cdot (Y_g - F_j^l) + \phi \cdot (rand - 0.5) & \omega > \omega_{N_f+m} \\ M_j^l + \alpha \cdot \chi_6 \cdot \left(\sum_{g=1}^{N_m} (\omega_{N_f+g} \cdot M_j^l) \Big/ \sum_{g=1}^{N_m} \omega_{N_f+g} \right) & else \end{cases} \tag{18}$$

Where the acceleration coefficients are denoted by χ_1, χ_2, χ_3, χ_4, χ_5, and χ_6. Furthermore, $\phi \cdot (rand - 0.5)$, the final phrase, is a random search. If the jth spider is in a great place, the search is less random. Therefore, as the number of iterations rises, the impact of randomness should diminish. To regulate this impact, an inertia weight is added, which is represented by Equation (19)

$$\omega = \omega_{max} - \frac{t}{t_{max}} \cdot (\omega_{max} - \omega_{min}) \tag{19}$$

Where ω_{max} and ω_{min} are the inertia weight's maximum and lowest values, respectively. The current and maximum iteration numbers are indicated by the symbols t and t_{max}, respectively. Therefore, Equations (3.3.2) and (21) represent the spider update in ISSO2.

$$F_j^{l+1} = \begin{cases} F_j^l + \alpha \cdot \chi_1 \cdot (Y_d - F_j^l) + \beta \cdot \chi_2 \cdot (Y_c - F_j^l) \\ \quad +\varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5) & \delta \leq PF \\ F_j^l - \alpha \cdot \chi_1 \cdot (Y_d - F_j^l) - \beta \cdot \chi_2 \cdot (Y_c - F_j^l) \\ \quad -\varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5) & else \end{cases} \tag{20}$$

$$M_j^{l+1} = \begin{cases} M_j^l + \alpha \cdot \chi_4 \cdot (Y_f - M_j^l) + \varphi \cdot \chi_5 \\ \quad \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5) & \omega > \omega_{N_f+m} \\ M_j^l + \alpha \cdot \chi_6 \cdot \left(\sum_{g=1}^{N_m} (\omega_{N_f+g} \cdot M_j^l) \Big/ \sum_{g=1}^{N_m} \omega_{N_f+g} \right) & else \end{cases} \tag{21}$$

A slight stage size also makes the spiders search more thoroughly, although the speed of convergence is comparatively slow. However, while spiders' searches are harsh and can result in the population of spiders missing the ideal location, a big search step size helps speed up convergence. Equation (22) defines ISSO3, which uses a linearly reduced search stage size to provide an equitable balance between computational accuracy and computation time.

$$\Delta S = \Delta S_{max} - \frac{t}{t_{max}} \cdot (\Delta S_{max} - \Delta S_{min}) \tag{22}$$

Where the minimum and maximum step sizes are denoted by ΔS_{min} and ΔS_{max}, respectively. Therefore, the following Equations (23) and (24) are used to update the spider locations in ISSO3.

$$F_j^{l+1} = \begin{cases} F_j^l + \Delta S \cdot \ [\alpha \cdot \chi_1 \cdot (Y_d - F_j^l) + \beta \cdot \chi_2 \cdot \left(Y_c - F_j^l \right) \\ \quad + \varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5)] & \delta \leq PF \\ F_j^l - \Delta S \cdot [\alpha \cdot \chi_1 \cdot (Y_d - F_j^l) - \beta \cdot \chi_2 \cdot (Y_c - F_j^l) \\ \quad - \varphi \cdot \chi_3 \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5)] & else \end{cases} \tag{23}$$

$$M_j^{l+1} = \begin{cases} M_j^l + \alpha \cdot \chi_4 \cdot (Y_f - M_j^l) + \varphi \cdot \chi_5 \\ \quad \cdot (Y_g - F_j^l) + \omega \cdot \phi \cdot (rand - 0.5)] & \omega > \omega_{N_f+m} \\ M_j^l + \alpha \cdot \chi_6 \cdot \left(\sum_{g=1}^{N_m} (\omega_{N_f+g} \cdot M_j^l) \Big/ \sum_{g=1}^{N_m} \omega_{N_f+g} \right) & else \end{cases} \tag{24}$$

To enhance the variety of the spider population at every iteration, dominant males and females engage in mating activities. As a result, ISSO4 performs the mutation process for non-dominant males, which is represented by Equation (25).

$$M_j^{l+1} = \begin{cases} M_{j1}^l + \eta \cdot (M_{j2}^l - M_{j3}^l) & \alpha > PM \\ M_j^l & else \end{cases} \tag{25}$$

The amplification coefficient for differential evolution is represented by η, and $j1$, $j2$, and $j3$ are three random numbers that differ from j. Equation (26) is used to get the mutation probability, which is shown by PM.

$$PM = \omega_{PM} \cdot \frac{K_j - K_{best}}{K_{worst} - K_{best}} \tag{26}$$

To increase the accuracy of classification, the ASSO module dynamically optimizes the important IRF hyperparameters, such as the number of trees, tree depth, and feature-subset ratio, by encoding them as spider positions in the search space. Every spider is a possible IRF configuration, and its fitness is determined by its classification accuracy (Equation (14)). ASSO corrects the position and chooses the global best (Y_g) with the highest accuracy. This active tuning mechanism optimizes the structure of IRF, minimizes misclassification and over-fitting, and dynamically adjusts the weights of features and decision thresholds depending on changing patterns of IoT threats.

The ASSO-IRF model has various benefits for IoT cybersecurity. It improves attack detection accuracy by improving feature selection with ASSO, which dynamically adjusts model parameters. The IRF enables reliable classification by distinguishing between valid and corrupted IoT devices. The method promotes real-time prediction based on adaptive learning, which makes it applicable to dynamic IoT settings. It minimizes false positives as it makes decision thresholds more precise and computes state transitions.

4 Result

The research aimed to determine the security and behavioural forecasting of the IoT devices in the control of the SecureNet-IoT framework in the network of connected devices. Experimental design and performance measures, such as precision, F1 score, recall, and accuracy, are used to evaluate the efficiency of the model in the detection of malicious cyber-threat events that can penetrate IoT systems.

4.1 Experimental Setup

To conduct performance testing in IoT cybersecurity, SecureNet-IoT uses a powerful hardware-software stack that can handle large amounts of information and make decisions in real-time. It is powered by an Intel Core processor, 16GB RAM, and a 500GB hard drive that is effective in managing data related to interactions with IoT devices, network traffic, and communication

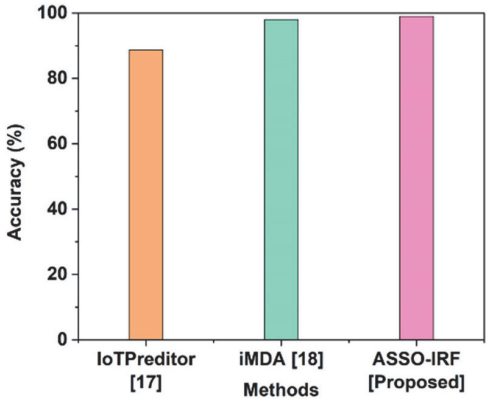

Figure 2 Graphical representation of accuracy.

logs. The software platform is developed on Ubuntu 20.04 and uses AI frameworks, including TensorFlow and Scikit-learn, to perform machine-learning operations.

4.2 Performance Metrics

The system performance is compared with the current system approaches to IoT Predictor [17] and IoT Malware Detection Architecture (iMDA) [18], and these are calculated on the basis of such metrics as F1 score, recall, accuracy, and precision.

• **Accuracy**

Accuracy is the number of cases in the IoT CS model classified correctly. It is described as providing an average performance metric, yet it can quite easily be deceptive if there is an imbalance in the dataset. This is relevant when measuring the extent to which SecureNet-IoT separates threats from normal device behavior within IoT security. A model that can accurately differentiate between genuine, breached, and counterfeit IoT devices has a high accuracy.

Table 2 Accuracy of system's performance using ASSO-IRF

Methods	Accuracy (%)
IoTPredictor [17]	88.68
iMDA [18]	97.93
ASSO-IRF [Proposed]	**98.96**

Table 3 Recall of system's performance using ASSO-IRF

Methods	Recall (%)
IoTPredictor [17]	94.95
iMDA [18]	88.73
ASSO-IRF [Proposed]	**96.24**

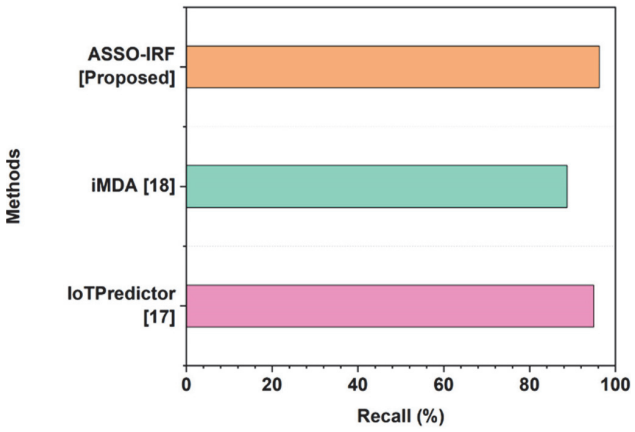

Figure 3 Graphical representation of recall.

Figure 2 and Table 2 illustrate the accuracy value of the ASSO-IRF model for evaluating the performance of the system, which has a higher accuracy of 98.96% as compared with the iMDA [18] of 97.93%.

- **Recall**

Recall gauges the system's capacity to accurately identify every instance of harmful activity in an IoT setting. A high recall lowers the likelihood of undiscovered attacks by guaranteeing that the majority of cyber-threats are recognized. It is crucial in SecureNet-IoT for early threat detection, as failing to identify an attack might risk network security.

Table 3 and Figure 3 represent the value of the recall of the ASSO-IRF model and indicate a high recall of 96.24% over the 94.95% of the IoT Predictor [17].

- **Precision**

Precision is used to measure the percentage of the harmful behaviors that have been accurately defined against the number of expected malicious behaviors. In further connection to IoT cybersecurity, false positives have to be reduced since they may wrongly identify normal devices as hostile. The high accuracy

Table 4 Precision of system's performance using ASSO-IRF

Methods	Precision (%)
IoTPredictor [17]	93.07
iMDA [18]	98.64
ASSO-IRF [Proposed]	**98.78**

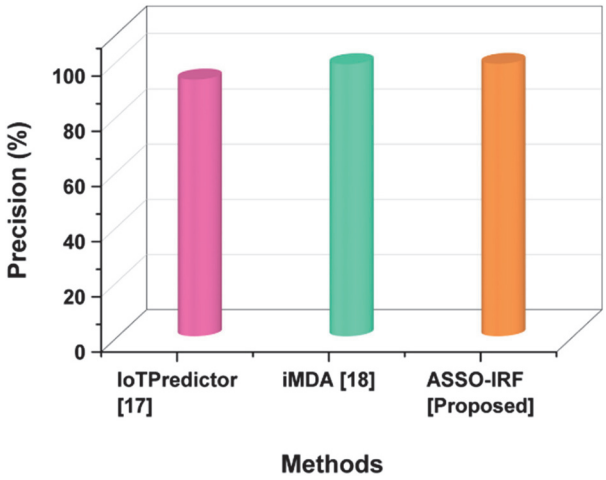

Figure 4 Graphical representation of precision.

of SecureNet-IoT minimizes the mislabeling rates of benign devices and, as a result, this enhances the effectiveness of the system in identifying the actual cyber-threats.

Table 4 and Figure 4 demonstrate the accuracy of the ASSO-IRF model, with a precision of 98.78 per cent compared with the 98.64 per cent shown with iMDA [18].

- **F1-score**

The F1-score is the harmonic mean of the recall and precision, which offers an accurate evaluation of the ASSO-IRF's performance. It performs particularly well in IoT CS on unbalanced datasets, minimizing both false negatives and positives. SecureNet-IoT has a high F1-score, which implies a perfect balance between identifying threats and minimizing false alarms. This metric ensures that the system is able to reliably classify IoT device statuses.

Table 5 and Figure 5 show the F1-score value of the ASSO-IRF model as a measure to assess the performance of the system, which possesses a high

Table 5 F1-score of system's performance using ASSO-IRF

Methods	F1-score (%)
IoTPredictor [17]	93.90
iMDA [18]	93.94
ASSO-IRF [Proposed]	**96.31**

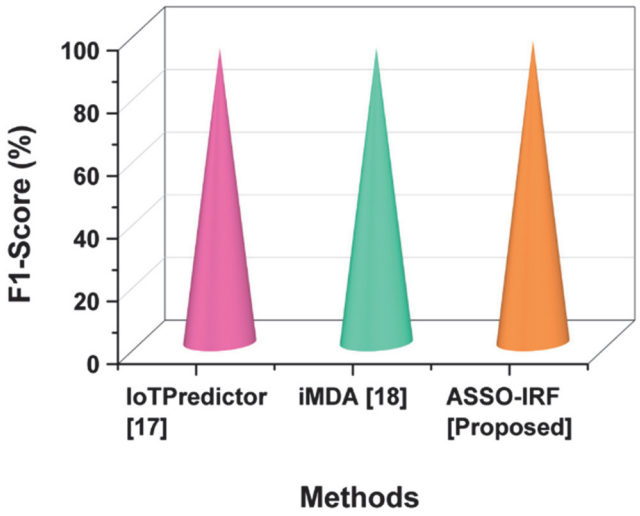

Figure 5 Graphical representation of F1-score.

value of F1-score equal to 96.31% when compared with that of iMDA [18] equal to 93.94%.

5 Discussion

A smart, optimization-based structure was developed in this research called SecureNet-IoT, to take threat detection, predict device behaviour, and classify anomalies to the next level in complicated IoT environments. Current architectures, including IoTPredictor [17] and iMDA [18], have severe flaws: IoTPredictor works with static parameters and cannot adapt to dynamic attacks, and iMDA, in turn, being highly accurate, has low recall, thus allowing undetected attacks and limiting generalization. The two methods do not have adaptive hyper-parameter optimization and real-time noise management, which result in over-fitting and non-scalability. To solve such problems, the suggested ASSO-IRF model combines the ASSO with an IRF, in which ASSO actively modulates hyperparameters, including tree depth,

feature-subsets ratio, and the estimation counts, by vibration-based search dynamics, and IRF by a kh2-based feature weight scheme to provide accurate classification. Such a combination of adaptive optimization and intelligent classification makes SecureNet-IoT able to adapt autonomously to changing behaviour of the IoT network, reduce false positives, and retain high detection accuracy.

6 Conclusion

The work reported in the research effectively established the design, development, and assessment of SecureNet-IoT, an enhanced security model aimed at increasing the network's security and IoT devices. The ASSO-IRF predicted activities for IoT devices and traced different attacks. SecureNet-IoT provides an assessment of improper behavior in IoT and fog computing environments, ensuring proactive identification and resolution of vulnerabilities by classifying devices as genuine, compromised, or counterfeit. The state change estimations that the framework applies would denote the early indications of a threat, whereas the correspondence tied between devices would grant increased reliability in the predictions imposed by SecureNet-IoT mechanisms. The findings showed that SecureNet-IoT operated well with high accuracy (98.96%), precision (98.78%), F1-score (96.31%), and recall (96.24%), hence improving the integrity and security of the IoT-assured systems. The latter is affected by high dynamism and low instance resources, which makes it difficult to apply to an IoT context with high dynamism. An extension to new technologies, including blockchain, that could improve security and scalability, may be a possibility in future research, since it is a place where SecureNet-IoT can be integrated.

Acknowledgements

The research is supported by: the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (22KJB580002), and the Excellent Teaching Team for QingLan Project of the Jiangsu Higher Education Institutions of China (Big Data Technology Teaching Team with Shipping Characteristic).

References

[1] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters, and A. Ng, "Cybersecurity data science: an overview from a machine learning perspective," *J. Big Data*, vol. 7, pp. 1–29, 2020. https://doi.org/10.1186/s40537-020-00318-5.

[2] P. Formosa, M. Wilson, and D. Richards, "A principlist framework for cybersecurity ethics," *Comput. Security*, vol. 109, p. 102382, 2021. https://doi.org/10.1016/j.cose.2021.102382.

[3] R. J. Raimundo and A. T. Rosário, "Cybersecurity in the Internet of Things in industrial management," *Appl. Sci.*, vol. 12, no. 3, p. 1598, 2022. https://doi.org/10.3390/app12031598.

[4] M. Kuzlu, C. Fair, and O. Guler, "Role of artificial intelligence in the Internet of Things (IoT) cybersecurity," *Discover Internet Things*, vol. 1, no. 1, p. 7, 2021. https://doi.org/10.1007/s43926-020-00001-4.

[5] H. El-Sofany, S. A. El-Seoud, O. H. Karam, and B. Bouallegue, "Using machine learning algorithms to enhance IoT system security," *Sci. Rep.*, vol. 14, no. 1, p. 12077, 2024. https://doi.org/10.1038/s41598-024-62861-y.

[6] S. Ahmed and M. Khan, "Securing the Internet of Things (IoT): A comprehensive study on the intersection of cybersecurity, privacy, and connectivity in the IoT ecosystem," *AI, IoT Fourth Ind. Revolut. Rev.*, vol. 13, no. 9, pp. 1–17, 2023.

[7] H. A. Alterazi et al., "Prevention of cybersecurity threats in IoT using particle swarm optimization," *Sensors*, vol. 22, no. 16, p. 6117, 2022. https://doi.org/10.3390/s22166117.

[8] D. K. Saini, H. Saini, P. Gupta, and A. B. Mabrouk, "Prediction of malicious objects using prey-predator model in IoT for smart cities," *Comput. Ind. Eng.*, vol. 168, p. 108061, 2022. https://doi.org/10.1016/j.cie.2022.108061.

[9] S. Strecker, W. Van Haaften, and R. Dave, "An analysis of IoT cybersecurity driven by machine learning," *Proc. Int. Conf. Commun. Comput. Technol. (ICCCT 2021)*, pp. 725–753, 2021. https://doi.org/10.1007/978-981-16-3246-4_55.

[10] M. Gopalsamy, "AI-based IoT-botnet attacks identification techniques to enhance cybersecurity," *Int. J. Res. Anal. Rev. (IJRAR)*, vol. 7, no. 4, pp. 414–420, 2020.

[11] D. Kalla, D. S. Kuraku, and F. Samaah, "Enhancing cybersecurity by predicting malware using supervised machine learning models," *Int. J.*

Comput. Artif. Intell., vol. 2, no. 2, pp. 55–62, 2021. https://doi.org/10.33545/27076571.2021.v2.i2a.71.

[12] A. Blowing, V. Stanislaw, R. Wagner, L. Ferrari, and S. Magomedov, "Performing ransomware detection through predictive behavioral mapping to autonomous threat identification," 2024. https://doi.org/10.31219/osf.io/5zu9r.

[13] Z. Umar et al., "Analysis of behavioral artifacts of malware for its detection using machine learning," *2024 IEEE 9th Int. Conf. Converg. Technol. (I2CT)*, pp. 1–5, Apr. 2024. https://doi.org/10.1109/I2CT61223.2024.10543310.

[14] Z. Jamadi and A. G. Aghdam, "Enhanced malware prediction and containment using Bayesian neural networks," *IEEE J. Radio Freq. Identif.*, 2024. https://doi.org/10.1109/JRFID.2024.3410881.

[15] N. U. Prince et al., "IEEE standards and deep learning techniques for securing IoT devices against cyber attacks," *J. Comput. Anal. Appl.*, vol. 33, no. 7, 2024.

[16] S. Rizvi, R. Pipetti, N. McIntyre, J. Todd, and I. Williams, "Threat model for securing IoT networks at the device level," *Internet Things*, vol. 11, p. 100240, 2020. https://doi.org/10.1016/j.iot.2020.100240.

[17] R. Kalaria, A. S. M. Kayes, W. Rahayu, E. Pardede, and A. Salehi, "IoT-Predictor: A security framework for predicting IoT device behaviors and detecting malicious devices against cyber attacks," *Comput. Security*, vol. 146, p. 104037, 2024. https://doi.org/10.1016/j.cose.2024.104037.

[18] M. Asam et al., "IoT malware detection architecture using a novel channel boosted and squeezed CNN," *Sci. Rep.*, vol. 12, no. 1, p. 15498, 2022. https://doi.org/10.1038/s41598-022-18936-9.

Biography

Yan Zhang, born in July 1984, Lecturer, Master's degree holder. She graduated from Nanjing University of Posts and Telecommunications with a major in Information Security, and currently serves as Director of the IoT Application Technology Professional Center at Jiangsu Maritime Institute. Her academic research focuses on IoT technology, network security technology, and electronic information technology. She has published 1 core Chinese paper in engineering and technology fields and 1 monograph.