Int. J. Appl. Math. Comput. Sci., 2026, Vol. 36, No. 3, 475–492
DOI: 10.61822/amcs-2026-0031

AN EFFICIENT ANOMALY DETECTION SYSTEM IN IoT EDGE USING CHI–SQUARE AND IMPROVED PARTICLE SWARM OPTIMIZATION FEATURE SELECTION WITH ENSEMBLE CLASSIFIERS

JACOB MANOKARAN [a,*], GURUSAMY VAIRAVEL [b], ARUNKUMAR GOPU [c], KAMEPALLI S.L. PRASANNA [d]

[a]Department of Electronics and Communication Engineering
Sri Krishna College of Engineering and Technology (SKCET)
Kuniyamuthur, Coimbatore, Tamil Nadu, 641 008, India
e-mail: manoraj3@gmail.com

[b]Department of Electronics and Communication Engineering
SRM Institute of Science and Technology
Irungalur, Tiruchirappalli, Tamil Nadu, 621105, India

[c]Department of Computer Science and Engineering
Dayananda Sagar University
Harohalli, Bengaluru, Karnataka 562112, India

[d]Department of Computer Science and Engineering
Amrita Vishwa Vidyapeetham
Mangalagiri, Kuragallu, Andhra Pradesh 522240, India

Anomaly detection using machine learning (ML) has become a critical research focus point in the modern digital era. Although recent ML-based models demonstrate strong detection capabilities, their performance is often limited by the vast volume and high dimensionality of data, leading to reduced accuracy, lower detection rates, and increased learning complexity. To address these challenges, this study proposes a novel ensemble learning framework that combines random forest (RF), extreme gradient boosting (XGBoost), adaptive boosting (AdaBoost), category boosting (CatBoost), and a light gradient boosting machine (LGBM) with a hybrid feature selection strategy. An improved particle swarm optimization (IPSO) algorithm enhanced with elimination and opposition-based learning is developed and integrated with the chi-square method (chi-IPSO) to optimize feature selection. Additionally, the synthetic minority oversampling technique (SMOTE) is employed to handle class imbalance in the datasets. The proposed framework is evaluated using two benchmark datasets, UNSW-NB15 and CICIDS 2017. Experimental results reveal that the RF algorithm with chi-IPSO achieves superior performance, attaining an accuracy of 94.58% on UNSW-NB15 and 99.70% on CICIDS 2017. These results illustrate the effectiveness of the proposed model, showing notable betterment over existing state-of-the-art anomaly detection approaches.

Keywords: high-dimensional data, chi-square, anomaly detection, ensemble learning, IPSO, feature selection, IoT security.

1. Introduction

The Internet of things (IoT) is one of the dominant communication paradigms that connects the physical and virtual worlds. It is revolutionizing the way people live and work. It extends to various domains, including smart cities, smart healthcare, smart banking, intelligent surveillance, and Industry 4.0 (Kumar *et al*., 2019). The IoT has a significant financial and community impact on people's lives. According to the IHS Markit report, the Internet will interlink over 100 billion IoT gadgets by 2030. Since the IoT has a large volume and heterogeneity

*Corresponding author

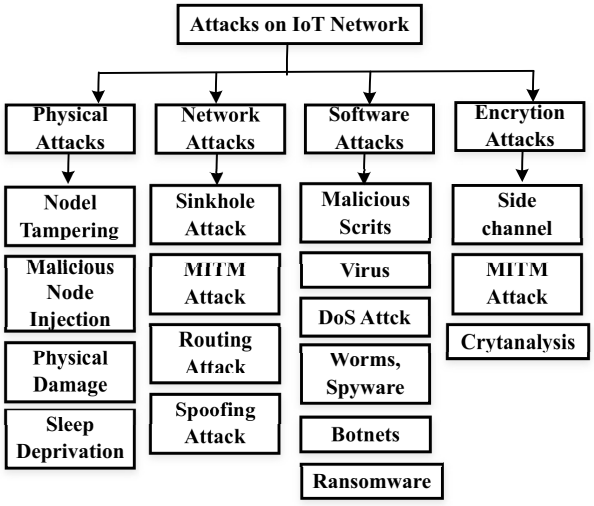


Fig. 1. IoT security attacks.

of devices, and is resource constrained, it has more security concerns than other sectors (Selvanarayanan *et al*., 2024). According to Kaspersky's statistics report, more than 15.37% of globally used Internet devices have experienced at least one attack. The 2023 global risks report identifies cyber-attacks as one of the top five sources of significant global risk (Kaur *et al*., 2023). Figure 1 highlights the recent attacks, which involve exploiting vulnerabilities in IoT devices.

In IoT network security, an intrusion detection system (IDS) is critical for sensing and identifying unauthorized access to compute resources and networks. Early detection of malicious behavior by IDSs allows prompt reactions and reduces potential damage. The IDS closely monitors system activity and network traffic, providing a comprehensive overview of the security situation. It also delivers real-time notifications upon detecting suspicious behavior, enabling a prompt action. It can assist in locating novel or unidentified threats, like zero-day attacks, that traditional detection techniques would overlook. Further, IDSs regularly update and upgrade their security measures to keep up with the dynamic nature of cyber-threats (Yalli *et al*., 2025).

There are two categories of IDSs: signature-based and anomaly-based. In the former, the data pattern is already stored in the database and matched with the input sequences for detecting anomalies. This method is simple and accurate, but it is not able to discover the zeroday attacks. The latter system type examines network traffic and compares it to an ideal pattern to detect anomalies. A significant advantage of this method is that it can predict new attacks (Manokaran and Vairavel, 2022b). Among the two types, we focus on anomaly-based detection methods.

Intrusion detection is a network analytic procedure in which ML algorithms are applied automatically to identify features of a user's abnormal or regular activity (Manokaran and Vairavel, 2022a). Recent ML algorithms have a high false alarm rate, which makes it tough to detect new types of attacks accurately. Combining several ML algorithms with weak predictive results generated through various projections of data, ensemble learning (EL) methods achieve better performance than any constituent approach by integrating results with different voting mechanisms (Manokaran and Vairavel, 2024). Tama and Lim (2021) conducted a structured mapping study of numerous EL algorithms used for abnormality detection. The authors concluded that these algorithms have better prediction accuracy than single-learning ones.

The huge volume and diversification of attacks in the open IoT edge environment make anomaly detection more complex. The multi-dimensional nature of the data reduces the performance of the IDS, increasing the computational and model-building time. Feature selection (FS) plays an energetic role in EL algorithms, expanding detection accuracy of the learning algorithm. FS techniques are categorized as filters, wrappers, and embedded methods (Maldonado *et al*., 2022). Filter methods are rapid and less computationally expensive, whereas the wrapper technique exhibits better performance.

1.1. Motivation and contributions. The fast growth in network traffic and the changing nature of cyber-threats require efficient anomaly detection systems that are both accurate and able to grow with the network. However, ML-based models that are already in use have issues such as high-dimensional data, class imbalance, and computational complexity, which make them less effective. This drives the creation of a strong ensemble-based framework that uses both feature selection and optimization strategies to get around these problems.

In our proposed system, we combined the benefits of filter and wrapper methods using the chi-IPSO algorithm for feature selection. Intergrating filter and wrapper techniques enables efficient dimensionality reduction while ensuring the selection of an optimal feature subset for improved accuracy. To explore the entire search space and evaluate possible attribute subsets, the chi-IPSO algorithm is used as a search tool. Initially, the imbalanced data are transformed into balanced using the SMOTE method. The model is trained with EL algorithms such as RF, the LGBM, AdaBoost, CatBoost, and XGBoost. Additionally, we enhance the performance of EL algorithms by tuning their parameters with the PSO technique. Another suggestive contribution of this study is that all the calculations occur at the edge of the device corner, which will increase the security and performance of the model. Figure 2 displays the developed framework for edge computing.

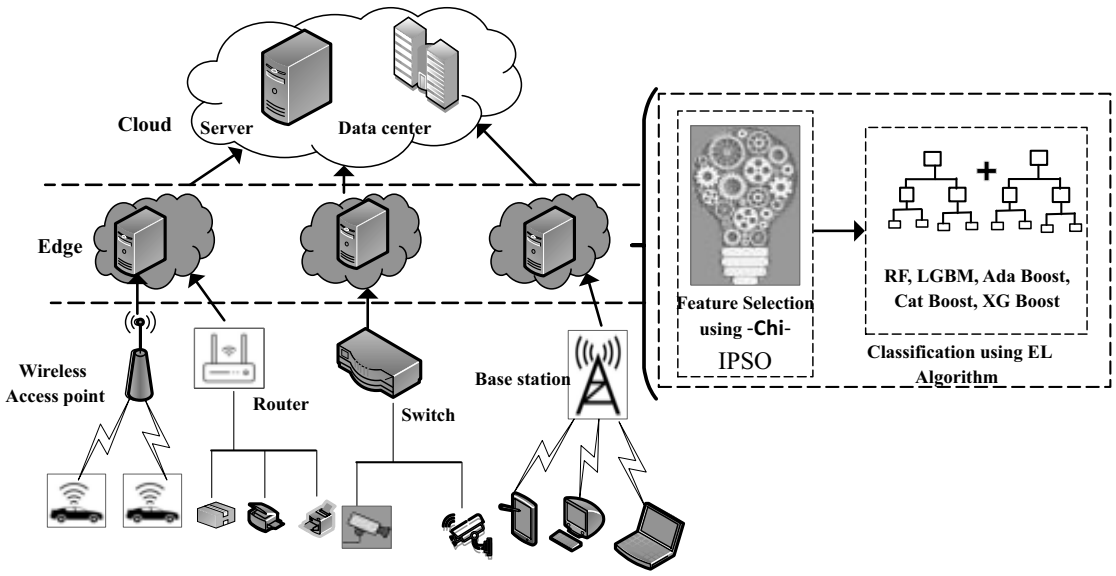


Fig. 2. Anomaly detection in IoT edge.

The suggested approach aims to improve classification accuracy while reducing the number of features and computational time. The key contributions of this study are as follows:

(i) The PSO algorithm is improved using elimination and opposition-based learning approaches and hybridized with the chi-square method.

(ii) The proposed chi-IPSO algorithm is used for the selection of optimum features in the anomaly detection model.

(iii) Then we use different PSO optimized EL algorithms such as RF, AdaBoost, the LGBM, CatBoost and XGBoost for network anomaly detection in an IoT scenario.

(iv) The developed model of the RF algorithm with chi-IPSO feature selection is compared with contemporary methods using accuracy, F1-score, and the MCC value.

The paper is organised as follows. Section 2 discusses recent similar studies on the use of optimization for an efficient IDS. Section 3 provides the proposed ADS and its parameters, and Section 4 presents the experimental setup of the developed system and results. Section 5 provides conclusions and discusses further work.

2. Literature review

In recent years, ML-based IDSs have been a central research area worldwide, and several models have been established to detect anomalies in the IoT. Fitni and Ramli (2020) improved the performance of an IDS using Spearman's rank coefficient-based feature selection and EL algorithms. Logistics regression (LR), decision tree (DT), and LGBM algorithms were chosen as the base classifiers and combined to attain 98.80% accuracy for the CICIDS 2018 dataset. The authors state that the classifier's hyper-parameters will be tuned and extended to a DL algorithm for large data prediction in future work. Albulayhi *et al*. (2022) modeled an IDS using ML algorithms and a novel feature selection one. Optimum feature sets were selected using the set theory concept for filter-based FS approaches (information gain (IG) and gain ratio (GR)). The authors conclude that, for the IoTIDS 20 dataset, the developed algorithm attained 99.70% accuracy. The implementation of the proposed algorithm on edge devices is suggested for future work.

Abdullah *et al*. (2018) enhanced the performance of ADSs using FS techniques and EL algorithms. The dataset was divided into multiple small sets, and IG feature selection was applied for each one. The RF, J48, and partial DT algorithms were applied for classification. The authors indicated that the AdaBoost algorithm can be used for FS and classification in the future. Mhawi *et al*. (2022) created an IDS using a hybrid FS method and the voting ensemble concept. Correlation feature selection coupled with forest panelized attributes was employed to select the attributes. The K-nearest neighbours (KNN),

support vector machine (SVM), RF, and naive Bayes (NB) algorithms were applied as the base algorithms and fused with majority voting concepts. The developed model achieved 99.70 % accuracy for the CICIDS 2017 dataset. In the future, more recent datasets may be used for evaluation.

Rahman *et al*. (2021) invented a novel FS technique for an IoT attack detection system. The SVM, NB, and C4.5 algorithms were applied as the base ML algorithms and merged using artificial neural networks (ANNs). The proposed method achieved 99.90% accuracy for the Aegean Wi-Fi Intrusion Dataset. Abu Alghanam *et al*. (2023) developed an IDS using improved pigeon-inspired optimization (PIO) feature selection with an EL algorithm. Local search combined with the PIO technique further enhances the selection of optimum features. The outcomes reveal that the advised method works better than other NIDS's (network intrusion detection system) strategies published in recent years.

Kaushik *et al*. (2023) compared the detection ability of various ML algorithms with FS techniques for IDSs. LR, RF, KNN, AdaBoost, DT, GrBoost, NB, XGBoost, and the SVM were the learning algorithms used. The chi-square and IG techniques were employed for attribute selection. RF with a feature selection algorithm achieved a precision value of 99.12%. Jaw and Wang (2021) produced an IDS using lightweight hybrid FS techniques and EL algorithms. Initially, genetic search and rule-based FS techniques were combined, and then the expectation-maximization SVM, k-means, and clustering algorithms were mixed using the majority voting principle. The proposed system attained 99.70% accuracy for the KDD Test-21 dataset.

In an effort to better utilize the potential of both paradigms for intrusion detection, Rajathi and Rukmani (2025) developed a hybrid learning model that integrates parametric and non-parametric classifiers. When compared to solo classifiers, the study showed that hybridization improved detection accuracy. Higher model complexity is the limitation of the work. A stacking EL technique that combined supervised and unsupervised models was also presented by Mills *et al*. (2024). Their model enhanced IDS capacity to generalize various attack patterns and heterogeneous IoT traffic behaviors. Umer *et al*. (2025) advance deep architectures by introducing a robust transformation-based IDS that uses stacked auto-encoders and transformer models. The authors employed a resampling technique that combines borderline-SMOTE and KNN undersampling to alleviate data skewness and guarantee class performance. Amouri *et al*. (2024) demonstrated the resilience of hybrid neural-ensemble solutions for IoT environments by combining XGBoost with DL in a hybrid model that achieved over 99% detection rates for the N-BaIoT dataset. Nassreddine *et al*. (2025) developed a novel

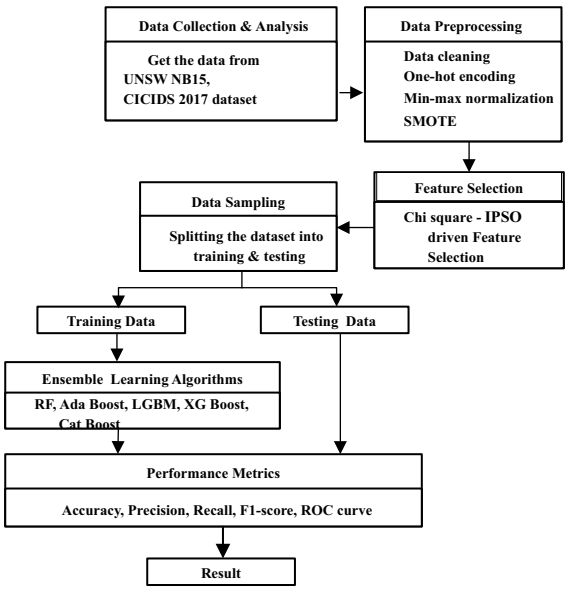


Fig. 3. Proposed anomaly detection framework.

approach that employs the EL algorithm by combining RF, DT, KNN, and XGBoost with LR as a meta-classifier. A hybrid FS method, integrating correlation-based filtering with XGBoost, is used along with SMOTE to handle data imbalance.

Das *et al*. (2022) performed a comparative analysis of NIDS's using ensemble feature selection and ML algorithms. Nine prominent FS techniques (recursive feature elimination (RFE), ANOVA, chi-square, LASSO, RF, mutual information, Pearson, LR, and SFPR) are combined using the majority voting principle and applied to the ensemble classifier. The developed algorithm attained an accuracy of 99.30% and a 0.5% error rate for the CICIDS 2017 datasets. Jiang *et al*. (2020) created a new IDS using the PSO-XGBoost algorithm. The PSO method is used to optimize the XGBoost algorithm's settings. For multi-classification, the parameter-optimized EL method has a high detection rate in comparison to the base EL algorithm.

From the related work, we observe that an optimum FS will increase the performance of the IDS, which is the aim of our work. For critical IoT scenarios, the performance of a single FS technique is not the appropriate approach, so we combine the filter and wrapper methods in our developed model.

3. Proposed methodology

This research aims to create an effective ADS using the chi-IPSO-based FS and optimized EL algorithms, which is shown in Fig. 3. Initially, the input data are gathered from online benchmark datasets: UNSW NB-15 and CICIDS 2017. Data scrubbing, encoding,

Table 1. Dataset distribution for our model analysis.

Traffic labels	UNSW-NB15		CICIDS 2017	
	Training	Test	Training	Test
Normal	74400	18600	363444	86318
Attack	131730	32932	89474	22368

and normalization are performed in the pre-processing stage. Then the SMOTE re-sampling technique is applied to balance an imbalanced dataset. Optimum features are chosen by a hybrid of chi-square feature selection and the IPSO algorithm. The dataset is split 80/20 between training and testing data using the hold-out method. Various EL algorithms like RF, XGBoost, AdaBoost, CatBoost and the LGBM are used to train and test the data. The final model's performance is assessed using standard performance metrics.

3.1. Data analysis. The nature of the information applied to train an EL algorithm significantly impacts the model's performance. Here, our suggested approach is trained and tested on two popular datasets: UNSW-NB15 (Moustafa and Slay, 2015) and CICIDS 2017 (Sharafaldin *et al.*, 2018). Recently, for IDSs, some other standard datasets have been available, like TON-IoT (Alsaedi *et al.*, 2020), SENMQTTset (Siddharthan *et al.*, 2022), and IoT-23 (Garcia and Erquiaga, 2020). These datasets have a limited number of features. Our primary goal is to improve the IDS's performance using optimal FS techniques. Furthermore, the majority of research papers in the literature test their developed models using these two datasets, hence we consider these two as well. The UNSW-NB15 dataset is available from the Australian Centre for Cyber Security (ACCS) at the University of New South Wales and focuses on network intrusions. It comprises nine attacks, including denial-of-service attacks, worms, backdoors, and fuzzes. The CICIDS 2017 dataset is provided by the Canadian Institute for Cybersecurity (CIC), University of New Brunswick. It has 77 features, 15 class samples, and seven attack categories. The dissemination of the data is presented in Table 1.

3.2. Data preprocessing. Pre-processing has the ability to accelerate an ML algorithm. Three steps are taken in the preparation stage: filtering, encoding, and normalization. In the filtering phase, unwanted and null information is altered by a threshold value. The categorical samples are changed into numeric ones in the data encoding stage by one-hot encoding. Further, the standard min-max technique normalizes the data in the preprocessing stage. The method uses linear adjustment to set lower and upper bounds for unnormalized data. The technique converts every extreme value component to 1, changes the base estimation of each element to 0, and adjusts the decimal values between 0 and 1 for other values. Finally, the conversion of data is calculated using

$$\alpha_{new} = \frac{\alpha_i - \alpha_{\min}}{\alpha_{\max} - \alpha_{\min}}, \tag{1}$$

where α_i is the original value in the dataset, $\alpha_{\min}$ is the minimum value in the dataset, $\alpha_{\max}$ is the maximum value in the dataset, and α_{new} is the normalized value, scaled between 0 and 1.

3.3. SMOTE. SMOTE is a data balancing method that generates synthetic samples for the minority class by interpolating between existing minorities instances, which helps to reduce class imbalance and improve classifier performance. Figure 4 illustrates the algorithm's working principle.

3.4. Feature selection (FS). FS is essential in model design, particularly in multi-dimensional IoT environments. Optimum FS can enhance its detection ability and decrease the model's training time. Compared to single feature selection techniques, hybrid approaches have better results. In our proposed system, we use a hybrid of the chi-square technique and the PSO algorithm, both of which are explained below.

3.4.1. Chi-square technique. The features in the training database are ranked using the $\tilde{\chi}^2$ technique to determine which are the most discriminative and have the highest detection accuracy (Thaseen *et al.*, 2018). Considering the attributes t_i and the class label c_j, $\tilde{\chi}^2$ is defined as

$$\tilde{\chi}^2(t_i, c_j) = \frac{N(F_1 Z - Y F_2)^2}{(F_1 + F_2)(F_1 + Z)(Z + F_2)(Y + Z)}, \tag{2}$$

where F_1, F_2 are the numbers of occurrences (frequency counts) of attribute t_i with class label c_j, Y is the number of occurrences of c_j appearing without t_i, Z is the number of occurrences of neither c_j nor t_i appearing together, N is the total number of samples in the dataset. To reduce dimensionality and retain only the most discriminative attributes, we employed a two-stage feature selection approach. In the first step, the chi-square statistical test was applied to rank features according to their dependency on the target class. Based on this ranking, the top 16 features were selected as a threshold to balance the trade-off between information retention and computational efficiency. Our intention behind choosing the top 16 features was to reduce dimensionality while retaining those attributes that showed relatively higher τ^2-statistics compared to the rest. Although the difference between the 16th and 17th ranked features is not very large, we considered 16 as a threshold to balance the trade-off between information retention and

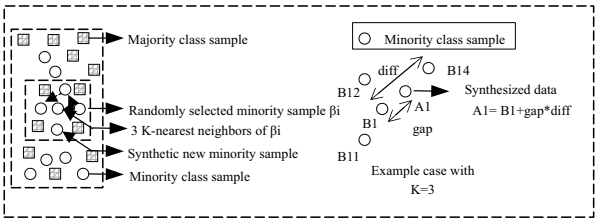


Fig. 4. SMOTE working principle.

computational efficiency. Figures 5 and 6 illustrate the chi-square scores of the features across different datasets (UNSW-NB15, CICIDS 2017). The chi-square value ranges from 0 to 1, with higher values indicating stronger correlation between the feature and the target class.

3.4.2. Particle swarm optimization (PSO) algorithm. PSO is a well-known intelligent searching technique developed in 1995 based on the principle of bird predation. It is zero-order and derivative-free. This implies that it is not dependent on gradients, making it applicable to a wide range of situations, including multi-modal and discontinuous problems. In PSO, the cluster fellows cooperate and share data to develop the best solution. By computing each bird's position and speed, a function value calculates the food density of each approaching location. In each search, the direction and speed of the investigation are adapted according to the difference between the best location for its history search and the best location for the population's history search. Ultimately, the entire bird swarm can assemble around the population's ideal site, leading to the discovery of the perfect solution (Louk and Tama, 2022). The processing steps of PSO are as follows:

(i) Random initialization of particle positions and velocity is done in the space of velocity and search.

(ii) Adjust the fitness function; then the $gbest$ is produced from the $pbest$ for each particle, each with a unique ideal solution. The outcome of the comparison will determine changes in the global optimal. Next, the historical global optimal is compared to the present global optimal.

(iii) The update of each particle's velocity (v) and position (x) is expressed as

$$v_{id} = wv_{id} + c_1 r_1([pbest_{id} - x_{id}] + c_2 r_2[gbest_{id} - x_{id}], \quad (3)$$

$$x_{id} = x_{id} + v_{id}, \quad i = 1, 2, 3, \ldots, Q,\ d = 1, 2, \ldots, T, \quad (4)$$

where $pbest$ is the individual particle's best value, d shows the dimension of particle i, $gbest$ is the global best value,

Algorithm 1. Improved particle swarm optimization (IPSO).

Input: N: population dimensions
P_i: Local optimal spot
P_g: Group optimal spot
F: fitness function
Output: P_g
Randomly set the spot α_i and velocity v_i of particle i
while criterion is not met **do**
  for $i = 1$ to N **do**
    Calculate the fitness value of each particle according to the fitness function
    for $i = 0;\ i < F_{\max};\ i{+}{+}$ **do**
      for $j = 0;\ j < E;\ j{+}{+}$ **do**
        Eliminate R worst swarms
        Generate R new swarms using Eqn. (6)
      end for
    end for
    Evaluate the fitness function
    if $F(\alpha_i) \geq F(P_i)$ **then**
      $P_i \leftarrow \alpha_i$
    end if
    if $F(\alpha_i) \geq F(P_g)$ **then**
      $P_g \leftarrow P_i$
    end if
  end for
end while
Update the position and velocity of particle i
Return P_g

w is the inertia factor, c_1, c_2 are the acceleration factors, and r_1, r_2 are the random numbers range from [0, 1].

3.4.3. Improved particle swarm optimization (IPSO) algorithm. PSO can considerably decrease the probability of reaching the local optimum prematurely while addressing the optimization issues. We need an enhanced PSO in order to strike the right balance between exploration and exploitation. We suggest an IPSO algorithm with two modifications, i.e., an elimination mechanism (EM) and the opposition-based learning method (OBL), to improve PSO's performance (Manokaran and Vairavel, 2023). To prevent the algorithm from entering the local best value, we first update the swarm group in accordance with the EM principle. After each algorithm iteration, we sort the fitness values corresponding to each swarm in ascending order and delete R swarms with the lowest fitness values. Meanwhile, the OBL method generates swarms equal to the number of eliminated ones.

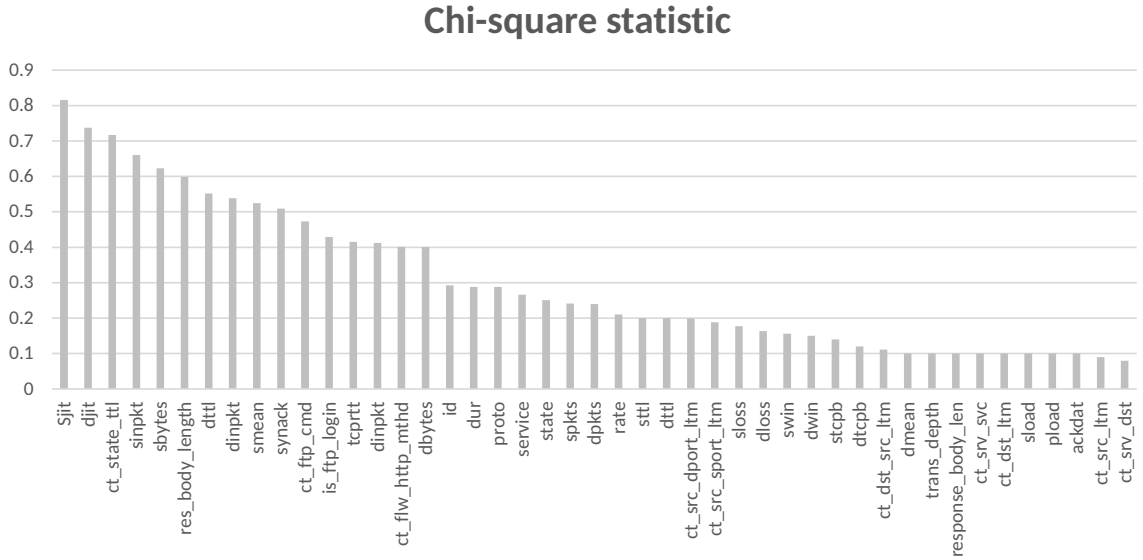


Fig. 5. Chi square statistic of the UNSW-NB15 dataset.

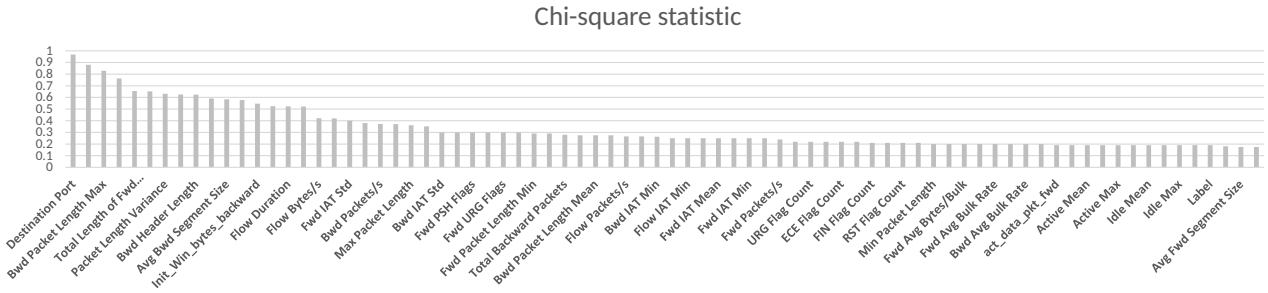


Fig. 6. Chi square statistic of the CICIDS 2017 dataset.

Figures 7 and Algorithm 1 display the IPSO algorithm's main flowchart and pseudo-code. In order to minimize the number of attributes and to enhance detection accuracy, we develop a fitness function. Using Eqn. (5), the fitness function is computed, where β_1 is a random weight factor falling between 0 and 1:

$$Fitness function = \beta_1 \times Acc + (1 - \beta_1) \times \frac{Selected features}{Total No. features}. \quad (5)$$

3.4.4. Feature selection using chi-square IPSO. Feature selection using chi-IPSO is implemented using two stages. The optimum features are ranked according to the chi-square method and applied to the input of the IPSO technique. The outcome of the IPSO algorithm is a sequence of 1's and 0's, where the attribute selection is denoted by one and attribute rejection is denoted by zero. The hybrid feature selection principle is shown in Fig. 9. The prominent 10 features selected using chi-IPSO are listed in Table 2. Regarding the observation that some features with higher τ^2-statistics were excluded by IPSO while those with relatively lower τ^2-statistics were retained, this highlights the complementary nature of our two-step process. The chi-square method performs an initial relevance-based filtering, while IPSO optimizes the final subset by considering multivariate interactions, redundancy, and classification performance. In this sense, IPSO may discard some features with individually high τ^2 scores if they are redundant in combination with others, while including features with lower τ^2 scores if they contribute positively in combination.

Algorithm 2. Chi-square IPSO method.

Input: Attributes, specification
Output: Best attributes
Step 1: Attributes = Chi-square (Attributes)
Step 2: Problem = Attributes
Step 3: Best attributes = IPSO (Problem, specification)
Step 4: Return best attributes

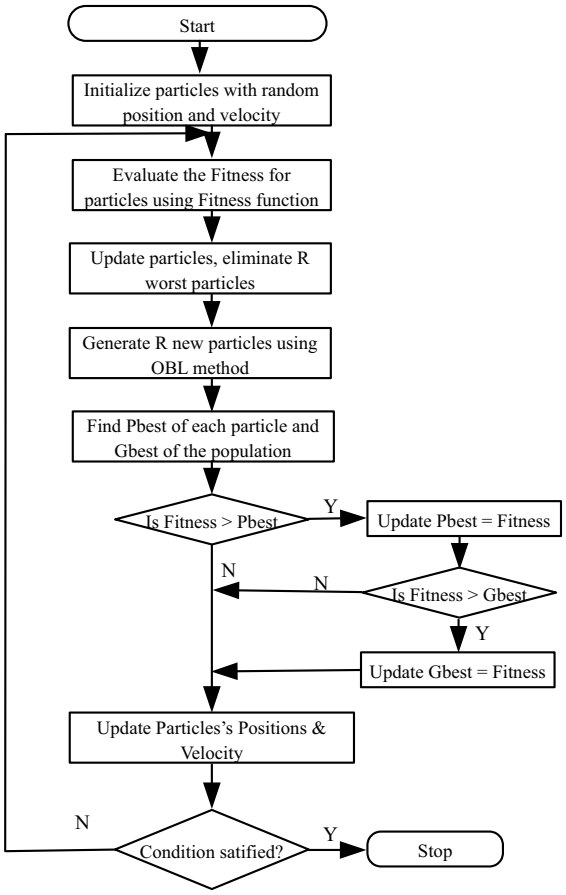


Fig. 7. Flowchart of the IPSO algorithm.

3.5. Ensemble learning algorithms. The ensemble learning algorithm boosts efficiency by breaking down a significant problem into numerous minor and superficial ones that are simpler to comprehend and solve using the divide-and-conquer method. In most situations, the EL algorithm performs better than the ML method. Our proposed system uses the following EL algorithms: RF, the LGBM, AdaBoost, CatBoost and XGBoost.

3.5.1. Random forest (RF). It is an EL algorithm used for efficient attacks and anomaly detection in the IoT (Manokaran *et al.*, 2023). Several DT methods are combined to create the RF algorithm, which produces a better result overall. Considering the input dataset having N samples and M features, the dataset is split into K numbers of new bootstrap datasets. The result obtained from the individual datasets is combined to produce the concluding result. The working structure is shown in Fig. 10.

3.5.2. Adaptive boosting (AdaBoost). AdaBoost combines several weak learners and produces strong

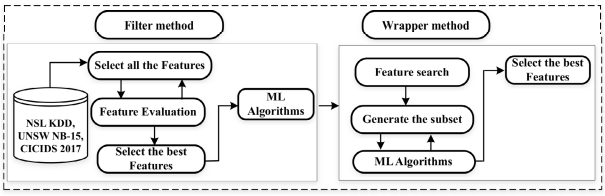


Fig. 8. Hybrid feature selection method.

Table 2. List of selected features using the chi-IPSO method.

UNSW-NB15	CICIDS 2017
smean	Total Length of Bwd Packets
tcprtt	Sub flow Fwd Bytes
dttl	Fwd Packet Length Max
sbytes	Bwd Packet Length Max
ct state ttl	Bwd Header Length
sinpkt	Init Win bytes backward
res body length	Fwd Packet Length Std
ct ftp cmd	Destination Port
dbytes	Packet Length Variance
din pkt	Init Win bytes forward

learners. The performance is boosted by iteratively changing the weight of the classifier. The performance will increase depending on the selected base classifier and the voting weight. The algorithm is shown as Algorithm 3 for the input dataset (α_1, β_1), (α_2, β_2),..., (α_n, β_n) (Jiang *et al.*, 2020).

3.5.3. Extreme gradient boosting (XGBoost). XGBoost is an EL algorithm used in various domains, like anomaly detection, attack detection, and feature selection problems. In the XGBoost algorithm, trees are expanded level-wise. The basic blocks of the algorithm are DT and gradient boosting. From the training data, a prediction is made for the probability of observing normal or anomaly data. The objective of the XGBoost algorithm is to select the best cost objective function, which is shown in Eqn. (6), where l is the loss function, $\widetilde{y_i}$ is a predicted value, t is the number of trees, α is a regularization term, and m is the number of samples in the training data (Jiang *et al.*, 2020):

$$\psi(\Omega) = \sum_{i=1}^{m} l(y_i, \widetilde{y_i}) + \sum_{j=1}^{t} \alpha(f_k). \tag{6}$$

3.5.4. Light gradient boosting machine (LGBM). The LGBM is a leaf-based algorithm used to raise trees vertically. The trees are expanded by considering where the loss is minimum during the splitting of a tree. Using a histogram-based method, the LGBM chooses the best split candidates. The significance

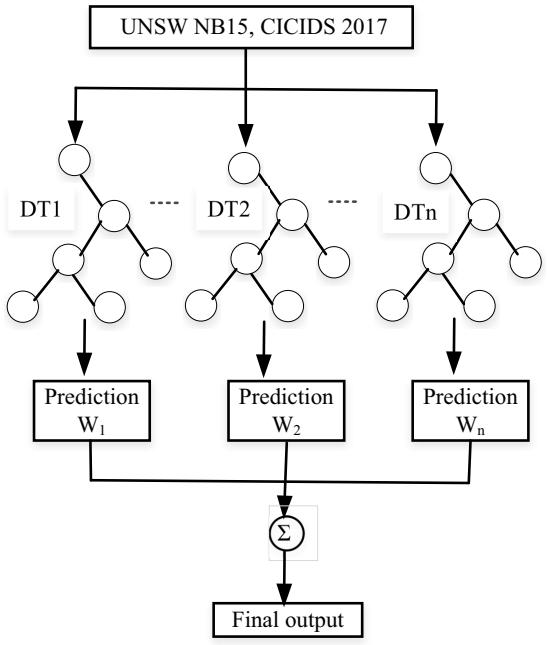


Fig. 9. Random forest tree splitting.

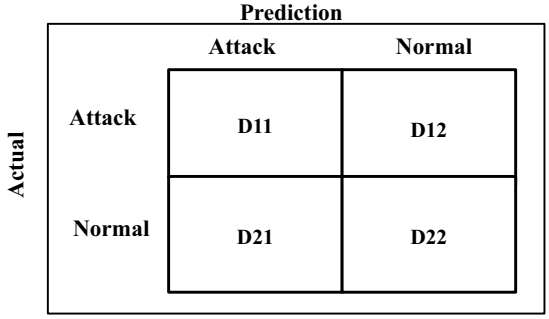


Fig. 10. Confusion matrix.

of data instances is highlighted by the LGBM using the sampling algorithm called gradient-based one-side sampling (GOSS) to improve training. Its main objective is to ignore minor gradients and concentrate on data samples with massive ones. Since data with short gradients have less errors, the underlying assumption is that they have already been trained. GOSS recommended eliminating these uninformative samples and using the remaining samples to calculate the knowledge obtained while finding the appropriate divides. Unfortunately, this will cause a bias issue in favor of the sample with more giant gradients and alter the initial data distribution. To address this issue, GOSS keeps all the samples with big gradients while randomly selecting the data with slight ones. For the purpose of computing the information gain, GOSS increases the weights of the data instances with small gradients (Alzamzami *et al*., 2020).

Algorithm 3. AdaBoost algorithm.

Transform weights $W_{et}(i) = 1/n; \quad i = 1, 2, \ldots, n$
For $t = 1, 2, \ldots, T_1$;
Using $W_{et}(i)$ to train a DT algorithm $D_t(\alpha)$
Select $D_t(\alpha)$ with low a weight error
$E_t = \sum_{i=1}^{n} W_{et}(i) I(y_i \neq D_t(\alpha))$
Specify $\lambda_t = \frac{1}{2} \ln \left[\frac{1-E_t}{E_t} \right]$
Update weight $W_{et+1}(i) = W_{et}(i) exp \left\{ \frac{-\lambda_t y_i D_t(\alpha_i)}{Z_t} \right\}$
Z_t: Normalization factor
Boosted output as follows: $H(\alpha) = \sum_{t=1}^{Total} \lambda_t D_t(\alpha_i)$

3.5.5. Category boosting (CatBoost). It is a powerful type of GBDT (gradient boosted decision trees) that has the ability to classify categorical variables automatically in various applications. It functions well with text, category and numerical data, and learns quickly. When compared to other algorithms, it is a prominent choice for GPU visualization.

3.6. Assessment parameters. The performance of the developed method is assessed using common ML algorithm evaluation metrics extrapolated from the confusion matrix. Figure 11 shows the matrix of our classification problem, where D11 denotes the number of anomalies that are correctly estimated, D12 stands for the number of anomalies incorrectly predicted, and D21 and D22 are the number of non-anomalies that were estimated incorrectly and correctly, respectively. Equations (7)–(11) list the evaluation parameters for our proposed model:

$$Accuracy = \frac{D11 + D22}{D11 + D22 + D12 + D21}, \tag{7}$$

$$Precision = \frac{D11}{D11 + D21}, \tag{8}$$

$$Sensitivity = \frac{D11}{D11 + D12}, \tag{9}$$

$$F1 = \frac{2 * Pre * Sen}{Pre + Sen}. \tag{10}$$

4. Result analysis

4.1. Experimental setup. This research uses a Lenovo system with Windows OS 64-bit. The PC has the Intel(R) Core i5 processor with 8 GB RAM. The attribute selection and ML algorithms are executed using Python with a Co-lab environment. Scikit Learn and the Pyswarm library are used in our experimental analysis. The developed model is trained and tested using a 20-fold CV method. The total of 30 experimental combinations are

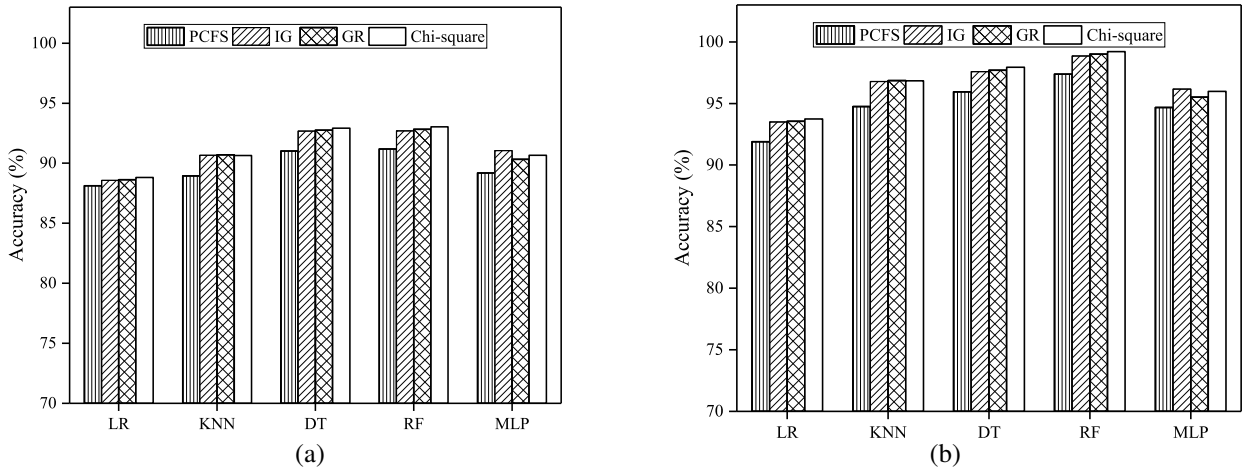


Fig. 11. Accuracy measure of filter-based FS techniques: UNSW-NB15 (a), CICIDS 2017 (b) datasets.

$$MCC = \frac{(D11 * D22) - (D12 * D21)}{\sqrt{(D11 + D21)(D11 + D12)(D22 + D21)(D22 + D12)}}. \tag{11}$$

Table 3. Initial setting of the PSO method.

S. no.	Parameters	Values
1	Number of populations	30
2	C_1, C_1	2
3	Inertia factor	0.7
4	Number of iterations	100

created with the use of five EL algorithms (RF, AdaBoost, the LGBM, CatBoost and XGBoost) and two intrusion datasets (UNSW-NB15 and CICIDS 2017).

Tama *et al*. (2020) performed an improved ADS in web traffic using EL algorithms. The authors proved that the performance of the ADS is improved by proper parameter-tuning of EL algorithms. PSO is used to detect the optimum hyper parameters of EL algorithms. The initial settings of PSO are shown in Table 3.

4.2. Discussion of results.

4.2.1. Performance analysis of filter-based FS techniques. The proposed model combine the chi-square and IPSO techniques for optimum feature selection.

Initially, we performed a comparative study of most prominent filter-based FS techniques such as the Pearson correlation coefficient, IG, the GR and the chi-square technique with an ML-based IDS. Among them the performance of chi-square technique is superior, so selected it for further studies. Figure 12 shows the performance comparison of filter-based FS techniques. From the figure, we observe that the performance of the chi-square method is better than that of the other methods.

4.2.2. Performance analysis of the EL algorithm with full features. We used five EL algorithms (RF, AdaBoost, the LGBM, CatBoost and XGBoost) with default parameters to compare the performance in this anomaly detection analysis. Tables 4 and 5 deliver the experimental assessment outcomes of the five EL-based anomaly detection results for various datasets with full features. The RF algorithm's performance is better than that of the remaining algorithms in terms of test accuracy, precision, sensitivity, F-score, and the MCC value for all datasets.

4.2.3. Performance comparison of the proposed model. Despite the increase in prediction accuracy, it is necessary to improve the detection accuracy of the EL algorithm. Hyper-parameter tuning is a crucial step to improve detection accuracy and save training time. The popular PSO algorithm enhances anomaly detection performance by fine-tuning the EL algorithm's parameters. Using the PSO method, we carefully tune several EL algorithm parameters, including tree size, the learning rate, and maximum depth, shown in Tables 6 and 7. Comparing Tables 4 and 5 as well as Tables 6 and 7, the performance of the tuned EL algorithm is better, and it is considered for further model design.

The developed model was used to conduct numerous experiments by tuning the number of features for each training session. Initially, all the features are ranked

Table 4. Detection results of full features with a default parameter EL algorithm on UNSW-NB15.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9268	0.9294	0.9590	0.6002	0.9588
LGBM	0.9378	0.9351	0.9711	0.6017	0.9651
XGBoost	0.9399	0.9365	0.9768	0.6050	0.9677
RF	0.9402	0.9371	0.9769	0.6075	0.9697
CatBoost	0.9390	0.9352	0.9768	0.6044	0.9659

Table 5. Detection results of full features with a default parameter EL algorithm on CICICDS 2017.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9790	0.9764	0.9810	0.9581	0.9787
LGBM	0.9799	0.9804	0.9788	0.9599	0.9796
XGBoost	0.9891	0.9886	0.9891	0.9781	0.9889
RF	0.9905	0.9907	0.9901	0.9811	0.9904
CatBoost	0.9878	0.9869	0.9888	0.9761	0.9878

Table 6. Detection results of full features with a tuned EL algorithm on UNSW-NB15.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9366	0.9316	0.9653	0.6013	0.9637
LGBM	0.9401	0.9474	0.9724	0.6108	0.9709
XGBoost	0.9432	0.9481	0.9732	0.6113	0.9724
RF	0.9434	0.9498	0.9812	0.6202	0.9817
CatBoost	0.9415	0.9483	0.9861	0.6110	0.9672

Table 7. Detection results of full features with a tuned EL algorithm on CICIDS 2017.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9802	0.9721	0.9853	0.9603	0.9886
LGBM	0.9880	0.9901	0.9912	0.9761	0.9922
XGBoost	0.9926	0.9925	0.9902	0.9852	0.9930
RF	0.9955	0.9955	0.9946	0.9911	0.9950
CatBoost	0.9911	0.9904	0.9917	0.9824	0.9929

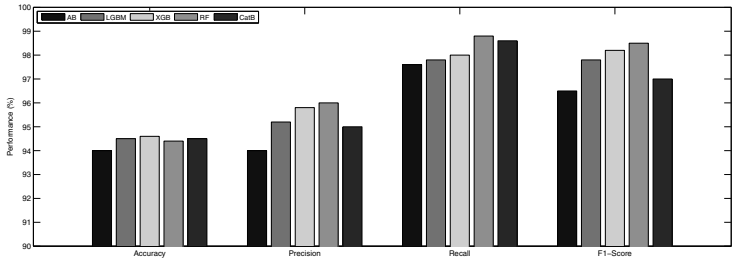
Table 8. Detection results of our proposed method on UNSW-NB15.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9394	0.9394	0.9764	0.6163	0.9642
LGBM	0.9441	0.9517	0.9752	0.6250	0.9763
XGBoost	0.9454	0.9574	0.9780	0.6275	0.9802
RF	0.9458	0.9590	0.9855	0.6328	0.9859
CatBoost	0.9447	0.9505	0.9881	0.6274	0.9690

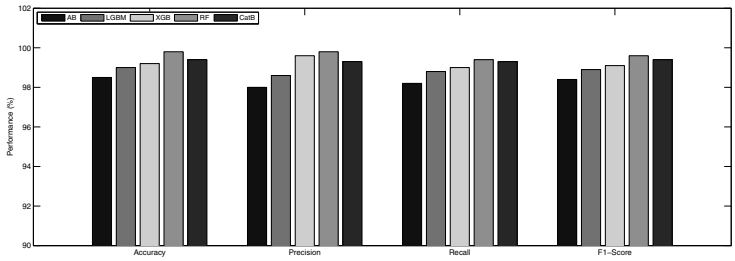
Table 9. Detection results of our proposed method on CICIDS 2017.

EL algorithm	Accuracy	Pre	Sen	MCC	F1-score
AdaBoost	0.9846	0.9812	0.9832	0.9692	0.9856
LGBM	0.9903	0.9932	0.9900	0.9806	0.9906
XGBoost	0.9940	0.9945	0.9912	0.9881	0.9934
RF	0.9970	0.9964	0.9932	0.9940	0.9955
CatBoost	0.9937	0.9929	0.9931	0.9875	0.9936

rapidly by applying the chi-square method. The top 16 most relevant ones are selected and applied as the input of the developed IPSO algorithm. After several iterations, we select ten most prominent characteristics for model building. Our proposed model chooses the critical features by combining the advantages of filtering

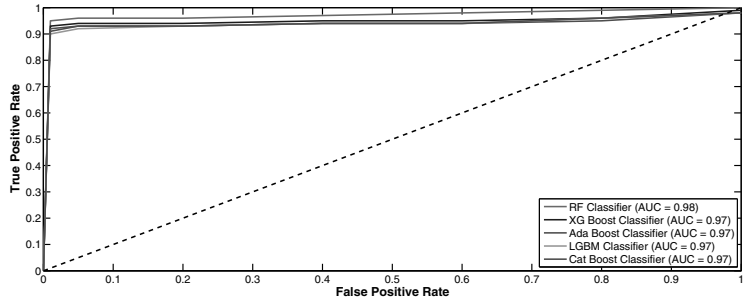


(a)

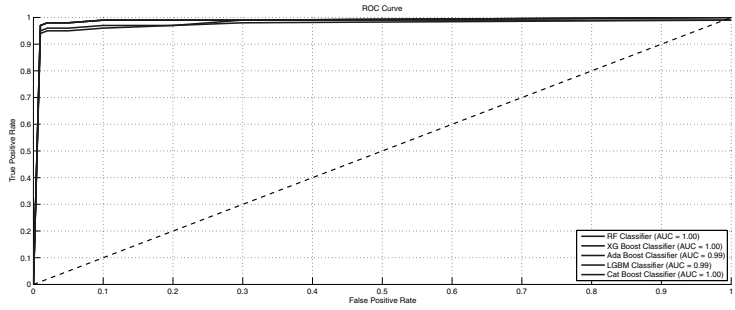


(b)

Fig. 12. Performance comparison of proposed model: UNSW-NB15 (a), CICIDS 2017 (b) datasets.



(a)



(b)

Fig. 13. ROC curve of the proposed model: UNSW-NB15 (a), CICIDS 2017 (b) datasets.

(chi-square) and the wrapper method (IPSO). Tables 8 and 9 show the detection results of the proposed methods (chi-IPSO-based feature selection) with an optimized EL algorithm on two datasets.

The performances of the UNSW-NB15 dataset with full features using the AdaBoost, LGBM, XGBoost, RF and CatBoost classifiers are 92.68%, 93.78%, 93.99%, 94.02%, and 93.90%, respectively. Similarly, the performances achieved with the feature selection technique using the AdaBoost, LGBM, XGBoost, CatBoost, and RF classifiers are 93.94%, 94.41%, 94.54%, 94.47%, and 94.58%, respectively. The

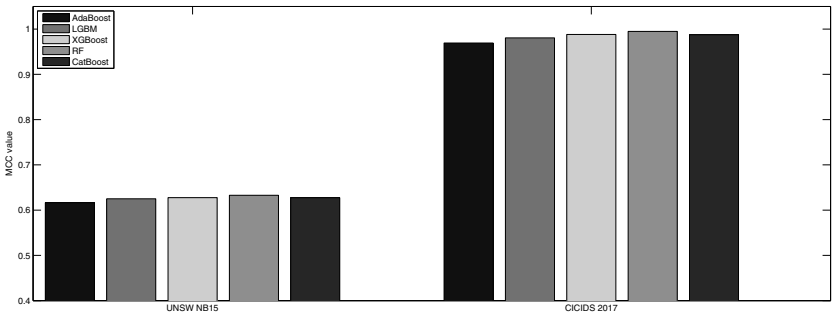
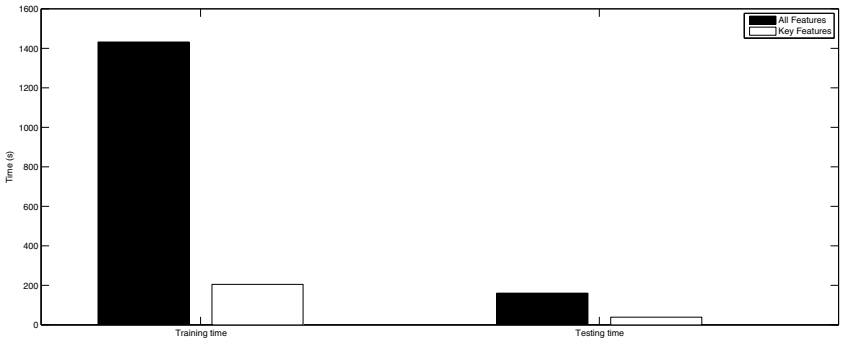


Fig. 14. MCC value comparison of different EL algorithms.



(a)



(b)

Fig. 15. Execution time comparisons: UNSW NB 15 (a), CICIDS 2017 (b) datasets.

performances of the CICIDS 2017 dataset without feature selection techniques using the AdaBoost, LGBM, XGBoost, CatBoost, and RF classifiers are 97.90%, 97.99%, 98.91%, 98.78%, and 99.05%, respectively. Similarly, the performances achieved with the feature selection technique using the AdaBoost, LGBM, XGBoost, CatBoost, and RF classifiers are 98.46%, 99.03%, 99.40%, 99.37%, and 99.70%, respectively. The above discussion infers that the performance of the classifiers with selected attributes is superior to that of the classifiers without the FS technique, irrespective of the dataset.

Figure 13 displays performance comparison of the developed model for the different datasets (UNSW-NB15, CICIDS 2017) using various evaluation metrics such as accuracy, precision, recall, and F-score. Clearly, the RF algorithm exhibits better performance compared to AdaBoost, the LGBM, CatBoost, and XGBoost for all datasets.

Figure 14(a) displays the ROC curve of the UNSW-NB15 dataset for several EL algorithms. The respective AUC values for each classifier are 0.97 (AdaBoost), 0.97 (LGBM), 0.97 (XGBoost), 0.97 (CatBoost), and 0.98 (RF classifier). It is evident that the RF algorithm has a superior AUC value among these classifiers. Figure 14(b) shows the ROC curve of various

Table 10. Effect of population size on RF performance with UNSW-NB15.

Particle size	Accuracy	Pre	Sen	F1-score
10	0.9058	0.9178	0.9455	0.9479
20	0.9268	0.9380	0.9645	0.9658
30	0.9458	0.9590	0.9855	0.9859
40	0.9460	0.9591	0.9854	0.9859

Table 11. Effect of population size on RF performance with CICIDS 2017.

Particle size	Accuracy	Pre	Sen	F1-score
10	0.9592	0.9588	0.9549	0.9598
20	0.9778	0.9772	0.9740	0.9763
30	0.9970	0.9964	0.9932	0.9955
40	0.9971	0.9965	0.9932	0.9955

Table 12. Effect of iteration count on RF performance with UNSW-NB15.

Iteration count	Accuracy	Pre	Sen	F1-score
50	0.9158	0.9278	0.9555	0.9579
75	0.9269	0.9385	0.9644	0.9655
100	0.9458	0.9590	0.9855	0.9859
125	0.9459	0.9590	0.9854	0.9859

Table 13. Effect of iteration count on RF performance with CICIDS 2017.

Iteration count	Accuracy	Pre	Sen	F1-score
50	0.9692	0.9688	0.9649	0.9698
75	0.9787	0.9779	0.9748	0.9767
100	0.9970	0.9964	0.9932	0.9955
125	0.9971	0.9965	0.9932	0.9955

EL algorithms for the CICIDS 2017 dataset. The AUC values for the individual classifiers are as follows: 0.99 (AdaBoost), 0.99 (LGBM), 1 (XGBoost), 1 (CatBoost), and 1 (RF classifier). Among these classifiers, the XGBoost, CatBoost, and RF algorithms have the best AUC values.

The MCC value comparison of the proposed chi-IPSO feature selection technique with EL algorithms for different datasets is shown in Fig. 15. For the UNSW-NB15 dataset, the MCC values for the different classifiers are: 0.6163 (AdaBoost), 0.6250 (LGBM), 0.6275 (XGBoost), 0.6274 (CatBoost), and 0.6328 (RF classifier). For the CICIDS 2017 dataset, the MCC values for the different classifiers are 0.9692 (AdaBoost), 0.9806 (LGBM), 0.9881 (XGBoost), 0.9875 (CatBoost) and 0.9940 (RF classifier). From the figure, clearly the RF algorithm has the highest MCC values of 0.6328, and 0.9949 for the UNSW-NB15 and CICIDS 2017 datasets. This means that the RF algorithm can forecast the result more accurately than the others. The proposed chi-IPSO FS technique reduces the number of features in the UNSW-NB15 dataset from 49 to 10, and from 79 to 10 for the CICIDS 2017 dataset. In light of this, the suggested model's training and testing duration is shorter than that of the complete features displayed in Fig. 16. In general, the more features that need to be analyzed, the longer the execution process takes.

4.3. Impact of particle population size. With the iteration count fixed at 100, the particle population was varied as 10, 20, 30, and 40. Smaller populations produced less-than-ideal solutions but converged more quickly. The balance between exploration and convergence was enhanced by increasing the population to 30, which resulted in higher accuracy. Upon particle population size of 30, the improvement in detection accuracy became marginal while computational cost increased significantly. The UNSW-NB15 and CICIDS 2017 datasets' performance with the RF algorithm is displayed in Tables 10 and 11.

4.4. Impact of iteration count. With a fixed particle size of 30, the number of iterations was varied to 50, 75, 100, and 150. Accuracy decreased at 50 iterations due to incomplete convergence. Up to 100 iterations, performance steadily improved. After the improvement in performance was negligible, indicating convergence of the optimization process. The UNSW-NB15 and CICIDS 2017 datasets' performance with the RF algorithm is displayed in Tables 12 and 13. The analysis demonstrates that the performance of PSO-based feature selection is greatly impacted by parameter tuning. The best detection accuracy is consistently obtained at a reasonable computational cost with an ideal configuration (particle size $= 30$, iterations $= 100$).

4.5. Comparative study. When the developed model is compared with the contemporary ADS, it can be identified to have various strengths and weaknesses, which enable further improvements. Kunhare *et al*. (2020) suggested an IDS using PSO with various ML algorithms; among them, RF+PSO produced an excellent detection rate of 99.26%. Dey *et al*. (2023) offered an efficient IDS using two new optimization techniques for FS, namely, the binary gravitational search algorithm and the GWO algorithm, and attained an accuracy of 99.41%. Additionally, we evaluated the developed chi-IPSO RF method by comparing it with other existing models and verified it on the benchmark dataset. Table 14 reveals that our proposed model is superior to most of the existing IDSs in the IoT.

5. Conclusions and future work

In this paper, we proposed an effective anomaly detection system in the IoT using chi-IPSO feature selection with

Table 14. Comparison the developed model with present IDS models.

Study	Dataset	Model	Accuracy
Abdullah *et al.*, 2018	NSL KDD	RF, DT	86.68%
Mhawi *et al.*,2022	CICIDS 2017	Voting (RF, SVM, NB, KNN)	99.70%
Das *et al.*, 2022	CIC-IDS 2017, NSL KDD,UNSW-NB15	Voting (DT, NB, LR, NN, SVM)	99.50%, 88.10%, 85.70%
Turukmane and Devendiran, 2024	UNSW-NB15	SMO-HPSO	94.12%
Liu *et al.*, 2021	UNSW-NB15	LGBM	85.89%
Roy *et al.*, 2022	NSL KDD	B-Stacking	98.50%
Yin *et al.*, 2023	UNSW-NB15	IGRF-RFE	84.24%
Henry *et al.*, 2023	CICIDS 2017	CNN-GRU	98.73%
Oyelakin, 2024	CICIDS 2017	XGBoost	98.00%
Muhammad *et al.*, 2023	CICIDS 2017	CI-EnsID	97.90%
Thockchom *et al.*, 2023	UNSW-NB15, CICIDS 2017	Stacking	93.88%, 99.80%
Thakkar and Lohiya, 2023	NSL KDD, UNSW-NB15, CICIDS 2017	DNN	98.90%, 96.70%, 98.74%
Ours	UNSW-NB15, CICIDS 2017	Chi-IPSO RF	94.58%, 99.70%,

optimum ensemble learning algorithms. The newly developed chi-IPSO algorithm selects dominant features from the IoT dataset. To address the shortcomings of basic PSO, which easily falls into the local optimum and has a slow convergence speed, we propose an IPSO algorithm, which is then hybridized with the chi-square method. The proposed chi-IPSO feature selection technique reduces the number of features in the UNSW-NB15 dataset from 49 to 10, and from 77 to 10 for the CICIDS 2017 dataset. Furthermore, we utilized various ensemble classifiers to examine classification errors. Finally, the simulation results reveal that the RF classifier with the proposed feature selection technique achieves the highest accuracy of 94.58% for UNSW-NB15 and 99.70% for CICIDS 2017. In the future, this study can be expanded using different feature selection techniques, novel balancing techniques, and deploying the DL-based model in real time to categorize network data.

References

Abdullah, M., Alshannaq, A., Balamash, A. and Almabdy, S. (2018). Enhanced intrusion detection system using feature selection method and ensemble learning algorithms, *International Journal of Computer Science and Information Security* **16**(2): 48–55.

Abu Alghanam, O., Almobaideen, W., Saadeh, M. and Adwan, O. (2023). An improved PIO feature selection algorithm for IoT network intrusion detection system based on ensemble learning, *Expert Systems with Applications* **213**: 118745, DOI: 10.1016/j.eswa.2022.118745.

Albulayhi, K., Abu Al-Haija, Q., Alsuhibany, S.A., Jillepalli, A.A., Ashrafuzzaman, M. and Sheldon, F.T. (2022). IoT intrusion detection using machine learning with a novel high performing feature selection method, *Applied Sciences* **12**(10): 5015, DOI: 10.3390/app12105015.

Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A. and Anwar, A. (2020). TON IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems, *IEEE Access* **8**: 165130–165150, DOI: 10.1109/access.2020.3022862.

Alzamzami, F., Hoda, M. and El Saddik, A. (2020). Light gradient boosting machine for general sentiment classification on short texts: A comparative evaluation, *IEEE Access* **8**: 101840–101858, DOI: 10.1109/access.2020.2997330.

Amouri, A., Al Rahhal, M.M., Bazi, Y., Butun, I. and Mahgoub, I. (2024). Enhancing intrusion detection in IoT environments: An advanced ensemble approach using Kolmogorov–Arnold networks, *2024 International Symposium on Networks, Computers and Communications (ISNCC), Washington DC, USA*, pp. 1–6, DOI: 10.1109/isncc62547.2024.10758956.

Das, S., Saha, S., Priyoti, A.T., Roy, E.K., Sheldon, F.T., Haque, A. and Shiva, S. (2022). Network intrusion detection and comparative analysis using ensemble machine learning and feature selection, *IEEE Transactions on Network and Service Management* **19**(4): 4821–4833, DOI: 10.1109/tnsm.2021.3138457.

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

Fitni, Q.R.S. and Ramli, K. (2020). Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems, *2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), Bali, Indonesia*, DOI: 10.1109/iaict50021.2020.9172014.

Garcia, S., Parmisano, A. and Erquiaga, M.J. (2020). IoT-23: A labeled dataset with malicious and benign IoT network traffic, Available online at `https://www.stratosphereips.org/datasets-iot23`.

Henry, A., Gautam, S., Khanna, S., Rabie, K., Shongwe, T., Bhattacharya, P., Sharma, B. and Chowdhury, S. (2023). Composition of hybrid deep learning model and feature optimization for intrusion detection system, *Sensors* **23**(2): 890, DOI: 10.3390/s23020890.

Jaw, E. and Wang, X. (2021). Feature selection and ensemble-based intrusion detection system: An efficient and comprehensive approach, *Symmetry* **13**(10): 1764, DOI: 10.3390/sym13101764.

Jiang, H., He, Z., Ye, G. and Zhang, H. (2020). Network intrusion detection based on PSO-XGBoost model, *IEEE Access* **8**: 58392–58401, DOI: 10.1109/access.2020.2982418.

Kaur, B., Dadkhah, S., Shoeleh, F., Neto, E.C.P., Xiong, P., Iqbal, S., Lamontagne, P., Ray, S. and Ghorbani, A.A. (2023). Internet of things (IoT) security dataset evolution: Challenges and future directions, *Internet of Things* **22**: 100780, DOI: 10.1016/j.iot.2023.100780.

Kaushik, B., Sharma, R., Dhama, K., Chadha, A. and Sharma, S. (2023). Performance evaluation of learning models for intrusion detection system using feature selection, *Journal of Computer Virology and Hacking Techniques* **19**(4): 529–548, DOI: 10.1007/s11416-022-00460-z.

Kumar, S., Tiwari, P. and Zymbler, M. (2019). Internet of things is a revolutionary approach for future technology enhancement: A review, *Journal of Big Data* **6**(1): 1–21, DOI: 10.1186/s40537-019-0268-2.

Kunhare, N., Tiwari, R. and Dhar, J. (2020). Particle swarm optimization and feature selection for intrusion detection system, *Sādhanā* **45**(1): 109, DOI: 10.1007/s12046-020-1308-5.

Liu, J., Gao, Y. and Hu, F. (2021). A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM, *Computers and Security* **106**: 102289, DOI: 10.1016/j.cose.2021.102289.

Louk, M.H.L. and Tama, B.A. (2022). PSO-driven feature selection and hybrid ensemble for network anomaly detection, *Big Data and Cognitive Computing* **6**(4): 137, DOI: 10.3390/bdcc6040137.

Maldonado, J., Riff, M.C. and Neveu, B. (2022). A review of recent approaches on wrapper feature selection for intrusion detection, *Expert Systems with Applications* **198**: 116822, DOI: 10.1016/j.eswa.2022.116822.

Manokaran, J. and Vairavel, G. (2022a). An empirical comparison of machine learning algorithms for attack detection in Internet of things edge, *ECS Transactions* **107**(1): 2403–2417, DOI: 10.1149/10701.2403ecst.

Manokaran, J. and Vairavel, G. (2022b). Smart anomaly detection using data-driven techniques in IoT edge: A survey, *in* V. Bindhu *et al.* (Eds), *Proceedings of the 3rd International Conference on Communication,Computing and Electronics Systems*, Springer, Singapore, pp. 685–702, DOI: 10.1007/978-981-16-8862-1_45.

Manokaran, J. and Vairavel, G. (2023). IGWO-SoE: Improved grey wolf optimization based stack of ensemble learning algorithm for anomaly detection in Internet of things edge computing, *IEEE Access* **11**: 106934–106953, DOI: 10.1109/access.2023.3319814.

Manokaran, J. and Vairavel, G. (2024). DL-ADS: Improved grey wolf optimization enabled AE-LSTM technique for efficient network anomaly detection in Internet of things edge computing, *IEEE Access* **12**: 75983–76002, DOI: 10.1109/access.2024.3405628.

Manokaran, J., Vairavel, G. and Vijaya, J. (2023). A novel set theory rule based hybrid feature selection techniques for efficient anomaly detection system in IoT edge, *2023 International Conference on Quantum Technologies, Communications, Computing, Hardware and Embedded Systems Security (iQ-CCHESS), Kottayam, India*, DOI: 10.1109/iq-cchess56596.2023.10391717.

Mhawi, D.N., Aldallal, A. and Hassan, S. (2022). Advanced feature-selection-based hybrid ensemble learning algorithms for network intrusion detection systems, *Symmetry* **14**(7): 1461, DOI: 10.3390/sym14071461.

Mills, G.A., Acquah, D.K. and Sowah, R.A. (2024). Network intrusion detection and prevention system using hybrid machine learning with supervised ensemble stacking model, *Journal of Computer Networks and Communications* **2024**(1): 1–25, DOI: 10.1155/2024/5775671.

Moustafa, N. and Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set), *2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia*, DOI: 10.1109/milcis.2015.7348942.

Muhammad, A., Murtza, I., Saadia, A. and Kifayat, K. (2023). Cortex-inspired ensemble based network intrusion detection system, *Neural Computing and Applications* **35**(21): 15415–15428, DOI: 10.1007/s00521-023-08561-6.

Nassreddine, G., Nassereddine, M. and Al-Khatib, O. (2025). Ensemble learning for network intrusion detection based on correlation and embedded feature selection techniques, *Computers* **14**(3): 82, DOI: 10.3390/computers14030082.

Oyelakin, A.M. (2024). A learning approach for the identification of network intrusions based on ensemble XGBoost classifier, *Indonesian Journal of Data and Science* **4**(3): 190–197, DOI: 10.56705/ijodas.v4i3.88.

Rahman, M.A., Asyhari, A.T., Wen, O.W., Ajra, H., Ahmed, Y. and Anwar, F. (2021). Effective combining of feature selection techniques for machine

learning-enabled IoT intrusion detection, *Multimedia Tools and Applications* **80**(20): 31381–31399, DOI: 10.1007/s11042-021-10567-y.

Rajathi, C. and Rukmani, P. (2025). Hybrid learning model for intrusion detection system: A combination of parametric and non-parametric classifiers, *Alexandria Engineering Journal* **112**: 384–396, DOI: 10.1016/j.aej.2024.10.101.

Roy, S., Li, J., Choi, B.-J. and Bai, Y. (2022). A lightweight supervised intrusion detection mechanism for IoT networks, *Future Generation Computer Systems* **127**: 276–285, DOI: 10.1016/j.future.2021.09.027.

Selvanarayanan, R., Rajendran, S., Algburi, S., Ibrahim Khalaf, O. and Hamam, H. (2024). Empowering coffee farming using counterfactual recommendation based RNN driven IoT integrated soil quality command system, *Scientific Reports* **14**(1), Article no. 6269, DOI: 10.1038/s41598-024-56954-x.

Sharafaldin, I., Habibi Lashkari, A. and Ghorbani, A.A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization, *Proceedings of the 4th International Conference on Information Systems Security and Privacy, Madeira, Portugal*, pp. 108–116, DOI: 10.5220/0006639801080116.

Siddharthan, H., Deepa, T. and Chandhar, P. (2022). SENMQTT-SET: An intelligent intrusion detection in IoT-MQTT networks using ensemble multi cascade features, *IEEE Access* **10**: 33095–33110, DOI: 10.1109/access.2022.3161566.

Tama, B.A. and Lim, S. (2021). Ensemble learning for intrusion detection systems: A systematic mapping study and cross-benchmark evaluation, *Computer Science Review* **39**: 100357, DOI: 10.1016/j.cosrev.2020.100357.

Tama, B.A., Nkenyereye, L., Islam, S.R. and Kwak, K.-S. (2020). An enhanced anomaly detection in web traffic using a stack of classifier ensemble, *IEEE Access* **8**: 24120–24134, DOI: 10.1109/access.2020.2969428.

Thakkar, A. and Lohiya, R. (2023). Attack classification of imbalanced intrusion data for IoT network using ensemble-learning-based deep neural network, *IEEE Internet of Things Journal* **10**(13): 11888–11895, DOI: 10.1109/jiot.2023.3244810.

Thaseen, I.S., Kumar, C.A. and Ahmad, A. (2018). Integrated intrusion detection model using chi-square feature selection and ensemble of classifiers, *Arabian Journal for Science and Engineering* **44**(4): 3357–3368, DOI: 10.1007/s13369-018-3507-5.

Thockchom, N., Singh, M.M. and Nandi, U. (2023). A novel ensemble learning-based model for network intrusion detection, *Complex Intelligent Systems* **9**(5): 5693–5714, DOI: 10.1007/s40747-023-01013-7.

Turukmane, A.V. and Devendiran, R. (2024). M-MultiSVM: An efficient feature selection assisted network intrusion detection system using machine learning, *Computers and Security* **137**: 103587, DOI: 10.1016/j.cose.2023.103587.

Umer, M., Tahir, M., Sardaraz, M., Sharif, M., Elmannai, H. and Algarni, A.D. (2025). Network intrusion detection model using wrapper based feature selection and multi head attention transformers, *Scientific Reports* **15**(1), Article no. 28718, DOI: 10.1038/s41598-025-11348-5.

Yalli, J.S., Hilmi Hasan, M., Tang Jung, L., Ibrahim Yerima, A., Adamu Aliyu, D., Danjuma Maiwada, U., Mahmood Al-Selwi, S. and Ur Rehman Shaikh, M. (2025). A systematic review for evaluating IoT security: A focus on authentication, protocols and enabling technologies, *IEEE Internet of Things Journal* **12**(12): 18908–18928.

Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F. and Kwak, J. (2023). IGRF-RFE: A hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset, *Journal of Big Data* **10**(1), Article no. 15, DOI: 10.1186/s40537-023-00694-8.



Jacob Manokaran received his BE in electronics and communication engineering at the Noorul Islam and College of Engineering, Kanyakumari, in 2009, and his ME in applied electronics at the Anna University of Technology, Tirunelveli, in 2011. He then completed his PhD at the SRM Institute of Science and Technology, Chennai, India. With 11 years of teaching experience in engineering colleges, he is currently serving as an assistant professor in the Department of Electronics and Communication Engineering at the SKCET—Sri Krishna College of Engineering and Technology, Coimbatore, India. His area of interests include the Internet of things, attack detection, machine learning, deep learning, edge computing, cluster computing, big data analytics. His list of academic achievements includes 10 publications as well as five journal and five conference papers.



Gurusamy Vairavel (IEEE Senior Member) received his bachelor's degree in electronics and communication engineering in 2001 from Madras University, India, his master's degree in communication systems in 2004 and his PhD degree in wireless communication in 2014 from Anna University, Chennai, India. He is currently serving as the dean for academics and educational initiatives, SRM Institute of Science and Technology, Trichy, India. His research interests are massive MIMO communication systems, antenna design, machine learning, the Internet of things and engineering education. He is presently serving as an editor or a guest editor for reputed journals and chaired reputed conferences. He is an active member of the American Society for Engineering Education, the TPACK Special Interest Group, and a fellow in the Institute of Engineers (India). Currently, he is in the member-at-large position in the CDIO international council. E-mail: `vairaveg1@srmist.edu.in`



Arunkumar Gopu is an accomplished academic and researcher with over a decade of experience in teaching, research, and academic leadership. He currently works as an associate professor at Dayananda Sagar University, Bangalore. His research interests span cloud computing, artificial intelligence (AI), data mining, multi-objective optimization, bio-inspired algorithms, and the Internet of things. He has made significant contributions to the field, with 18+ publications, including SCI-indexed journal articles, conference proceedings, and book chapters. E-mail: `arunkumar.goge@gmail.com`



Kamepalli S.L. Prasanna is an assistant professor in the Department of Computer Science and Engineering, School of Computing, at Amrita Vishwa Vidyapeetham, where she has been working since February 2024. She completed her PhD from VIT-AP University in 2024. She received her MTech degree in 2016 and her BTech degree in 2011, both in computer science and engineering from from Jawaharlal Nehru Technological University. Her current research interests include soft computing, machine learning, and deep learning. E-mail: k_prasanna@av.amrita.edu

Received: 10 June 2025
Revised: 1 October 2025
Re-revised: 24 December 2025
Accepted: 8 January 2026