ASTES

Advances in Science, Technology and Engineering Systems Journal
Vol. 11, No. 2, 17-29 (2026)
www.astesj.com

ASTESJ
ISSN: 2415-6698

Hybrid Feature Selection for Anomaly Detection in IoT Network Intrusion Detection Systems

Mya Soe Soe Moe*[1], Win Mar Oo[2]

[1]*Faculty of Computer Science, University of Computer Studies Mandalay, Myanmar*

[2]*University of Computer Studies Mandalay, Myanmar*

Email(s): myasoesoe.moe@gmail.com (M. Moe), winmaroo.ucs@gmail.com (W. Oo)

∗Corresponding Author: Mya Soe Soe Moe, Faculty of Computer Science, University of Computer Studies Mandalay, 05071, Myanmar, Contact No & Email: myasoesoe.moe@gmail.com

ARTICLE INFO

Article history:
Received: 26 February, 2026
Revised: 15 March, 2026
Accepted: 17 March, 2026
Online: 5 April, 2026

Keywords:
IoT security
Intrusion detection system
Hybrid feature selection
Anomaly detection
Machine learning
Network traffic

ABSTRACT

The rapid growth of Internet of things (IoT) devices have heightened the need for effective Intrusion Detection System (IDS) to combat evolving cyber threats. The IoT networks has the security challenges due to the heterogeneous and high-dimensional nature of network traffic data, redundant features, and class imbalance which hinder detection accuracy and efficiency. Effective IDS requires robust feature selection mechanisms to enhance detection accuracy and reduce computational complexity. The research proposes a hybrid feature section method that combines filter and embedded techniques through a weighted scheme. Chi-square and Mutual Information scores are fused with a weighting mechanism and interests with Random Forest feature importance for anomaly detection in IoT environments. The proposed hybrid feature selection approach is evaluated on three benchmark IoT intrusion detection datasets, ACI-IoT2023, BoT-IoT, and TON-IoT datasets using five supervised classifiers: Random Forest, Decision Tree, K-Nearest Neighbour, Support Vector Machine, and Naïve Bayes classifiers. The study evaluated the proposed system for both binary and multiclass classification scenario. Experimental results demonstrated that the proposed feature selection with Random Forest classifier achieves the highest performance 99.93%, 99.98%, 99.01% in accuracy on each dataset respectively.

1. Introduction

The rapid proliferation of Interne of Things (IoT) devices has transformed modern communication networks by enabling seamless connectivity among heterogeneous smart objects. With the growth of IoT networks and the Internet, the devices are increasingly deployed to analyze data in critical application domains such as smart cities, healthcare, industrial automation, and intelligent transportation systems[1]. IoT networks are highly vulnerable to a wide range of cyberattacks due to their distributed architecture, limited computational resources, and lack of robust security mechanisms[2]. Therefore, Intrusion Detection Systems (IDSs) have become a necessary addition to the security infrastructure of internetworking environment.

The IDS systems analyse the network traffic data with related a large number of features by monitoring network traffic and discover unauthorized intrusions, identifying malicious activities. Among various IDS approaches, anomaly-based detection methods have gained considerable attention due to their ability to detect previously unseen and zero-day attacks. The effectiveness of anomaly-based IDS largely depends on the quality of the input features extracted from network traffic data. IoT intrusion datasets are typically high-dimensional and contain redundant or irrelevant features, which can degrade detection accuracy, increase false alarm rates, and impose significant computational overhead [3].

Feature selection has emerged as a crucial preprocessing step for improving the performance and efficiency of IDS. Conventional feature selection techniques can be broadly categorized into filter, wrapper, and embedded methods. Filter-based methods, such as chi-square and mutual information, are computationally efficient and independent of learning algorithms, but fail to capture complex feature interactions. Wrapper and embedded methods, on the other hand, consider classifier performance during feature selection and generally provide better

accuracy, albeit at higher computational costs. Relying on a single feature selection strategy may not adequately address the diverse characteristics of IoT network traffic [4].

To overcome the limitations, hybrid feature selection approaches have been increasingly explored to combine the advantages of multiple feature selections. In the study, a hybrid feature selection framework is proposed that integrates chi-square and mutual information scores to evaluate feature relevance, followed by an embedded feature selection mechanism based on Random Forest feature importance. The fusion of statistical dependency and information-theoretic measures is used to eliminate the irrelevant and redundant features which enhance the performance of machine learning models in detecting anomalies within IoT networks. The contributions of the system are as follows:

- A hybrid feature selection framework is designed to identify a compact and informative subset of features.
- It incorporates chi-square and mutual information scores for effective feature ranking and integrate an embedded Random Forest (RF)-based feature importance mechanism to refine the selected feature subset.
- The system uses a feature weighting mechanism that prioritize both statistical relevance and contextual importance, thereby enhancing the accuracy and performance of anomaly detection.
- The evaluation conducts with three IoT datasets that are balanced and imbalanced real-world intrusion data.
- The system reduces the feature dimensionality, thereby decreasing training time while maintaining high detection accuracy.

The structure of the paper is organized as follows: Section 2 discusses the previous related work on intrusion detection system and feature selection. Section 3 outlines the background theories of the study and presents the proposed system including feature selection algorithm. Section 4 obtains a detail description of the datasets. Section 5 presents discussion of the experimental results. Section 6 concludes the paper and highlights future directions.

2. Related Work

Numerous studies have been conducted to classify anomalies in IoT infrastructure using machine learning techniques. The research in [5],the author proposed an IDS model based on deep learning, integrating Pearson Correlation Coefficient with Convolutional Neural Network (PCC-CNN) to detect network anomalies. The model achieved the accuracy of 98%, 99%, and 98% on the three datasets. However, the PCC-based feature selection method cannot directly handle categorical variables and fails to capture interactions between features.

In the research [6] , the author presented a model for intrusion detection in the IoT-based edge computing using CNN. The study tested on the CICDDoS2019 dataset. It achieves the high performance of 99.68% accuracy for binary classes, 99.90% for 8-classes, and 99.95% for 13-classes when identifying various types of DDoS attacks. The model is only trained and tested on one dataset. Therefore, the system reduces generalizability to real-world IoT environments with different attack patterns.

An IoT intrusion detection system is designed in [7] to address the growing security challenges arising from the rapid expansion and inherent vulnerabilities of IoT technologies. For the ToN-IoT network dataset, the authors conducted the performance evaluation of ten learning algorithms including both base and ensemble classifiers. The proposed stacking ensemble model, which integrates CatBoost, Extra Trees, and XGBoost, demonstrates improve detection capability in both binary and multiclass intrusion detection scenarios. Experimental results indicate exceptionally high Matthews correlation coefficient (MCC) values of 0.9971 for binary classification and 0.9909 for multiclass classification. The framework achieved improved performance in heterogeneous IoT environments against sophisticated cyberattacks.

An advanced intrusion detection framework is proposed in [8] for IoT networks that combines quantum-inspired optimization, neuro-fuzzy feature selection, and multi-stage classification to address the growing complexity of IoT security threats. By integrating Quantum-Inspired Particle Swarm Optimization (QIPSO) with Adaptive Neuro-Fazzy Inference System (ANFIS). The study performed optimized feature selection by reducing data redundancy and enhancing detection accuracy. In the classification part, the system used Capsule Networks and Attention-Enhanced Recurrent Neural Network to capture both hierarchical feature relationships and temporal traffic patterns in detection of subtle and sophisticated attacks. Experimental validation achieved the performance of 98.83% accuracy, 98.56% precision, 98.65% F-measure on ION-IoT dataset and 98.6% accuracy, 98.5% precision, 98.94% F-measure on BOT-IoT dataset. The framework exhibited strong detection capability and robustness against diverse attack types, the study acknowledges limitations related to adversarial vulnerability and computational overhead, which hinders real-time deployment in resource-constrained IoT environments.

The intrusion detection system to address the challenges of high-dimensional data, dataset imbalance, and resource constraints in IoT networks using a hybrid handcrafted feature selection approach [9]. The important features are selected using Mutual Information and Sequential Feature Selector and then combine the two feature subsets. To test the performance of feature selection approach, four machine learning algorithms are employed. The study reduced feature redundancy and selected high critical traffic attributes for each traffic instance that enhances intrusion detection efficiency while maintaining low computational overhead. Evaluation results achieved an accuracy of 98.97%, 99.99% and 98.97%, 99.53% F1-score on NSL-KDD dataset and BOT-IoT dataset respectively. The model exhibited low latency and high throughput suitable for real-time intrusion

detection in large-scale IoT environments. The authors only evaluated on binary class as attack or non-attack.

In the study [10], deep learning-based intrusion detection for IoT environments to address key security challenges such as device heterogeneity, limited computational resources, dynamic traffic behaviour, and the growing threat of IoT botnets. By using convolutional neural networks, the approach captures both spatial and temporal characteristics of IoT network traffic. The evaluation is conducted using the UQ-NIDS and BoT-IoT datasets which are representative of real-world IoT intrusion and botnet scenarios. The study emphasized the critical role of data pre-processing, incorporating both instance-based techniques for data cleansing and feature-based techniques such as transformation, normalization, and dimensionality reduction to enhance IDS performance. Experimental result attained 84% in accuracy, precision, recall, and 82% in F1-score using Multilayer Perceptron for multiclass classification scenario.

In the study [11], the authors presented the introduction detection system for BoT-IoT dataset to address the limitations of existing datasets that often lack detailed attack scenarios, precise labelling, and representation IoT traffic. The study incorporated both legitimate and simulated IoT network traces alongside diverse and contemporary attack types. The feature selection process provided 9 features to reduce the input variables required for training and testing. The authors assessed the reliability and effectiveness of the BoT-IoT dataset using six machine learning approaches: Gradient Boosting, Artificial Neural Networks, Decision Trees, Logistic Regression, Naïve Bayes, and Random Forest classifier. Experimental results demonstrated that Random Forest classifier achieved high performance than others.

Despite significant progress in intrusion detection system for IoT networks, the critical gaps remain in existing research. The researches focus on achieving high accuracy metrics but provide limited analysis of latency, throughput, energy consumption, and robustness against adversarial manipulation which are essential for real-world IoT application. Dataset imbalance and insufficient representation of emerging and stealthy attacks, particularly IoT-specific botnets, remain unresolved changes.

3. Proposed Methodology

A hybrid feature selection framework is developed to enhance anomaly detection performance in IoT network intrusion detection systems. The approach is to identify a compact and highly discriminative feature subset from high-dimensional IoT traffic data while maintaining computational efficiency.

Figure 1 adapted from [9] presents the overall architecture of the proposed IoT network intrusion detection system. The system includes three main stages such as data preprocessing, hybrid feature selection and classification. The output is attacking type in binary-class or multi-class.

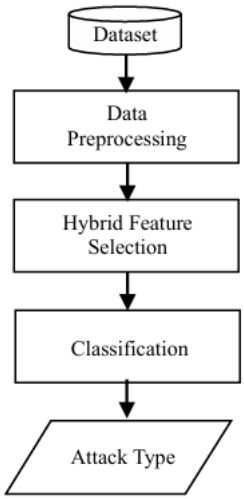

Figure 1: Overall architecture of the proposed system

3.1. Data Preprocessing

Data preprocessing is a crucial step in the development of reliable intrusion detection systems. Raw IoT intrusion datasets contain categorical attributes, varying feature scales, and highly imbalanced class distributions, which can adversely affect the performance of machine learning models. A comprehensive pre-processing strategy is applied in the study including label encoding of categorical features, feature normalization, and data balancing.

3.2. Label Encoding

IoT network traffic datasets include categorical features such as protocol types, service identifiers, and attack categories that cannot be directly processed by machine learning algorithms. To transform these categorical attributes into numerical representations, label encoding is employed in the study. Each distinct category is mapped to a unique integer value, enabling efficient computation while preserving the discrete nature of the original data. The approach ensures compatibility with the feature selection techniques and classifier used in the proposed framework.

3.3. Feature Normalization

The features extracted from IoT network traffic data exhibit different numerical ranges and scales, which can adversely affect the performance of the system. Min-Max normalization is applied in the study to rescale all numerical features into a fixed range between 0 and 1. The technique preserves the original distribution of the data while ensuring that all features contribute equally during model training and feature selection. By eliminating scale-related bias, the normalization improves model stability, convergence, and support fair evaluation in the proposed hybrid feature selection framework[12].

3.4. Data Balancing Using SMOTE

IoT intrusion detection datasets are characterized by severe class imbalance, where normal traffic instances significantly

outnumber attack samples or certain attack classes are underrepresented. The Synthetic Minority Over-Sampling Technique (SMOTE) is employed to balance the class distribution in the training data [13]. SMOTE generates synthetic samples for minority classes by interpolating between existing instances, thereby reducing bias toward majority classes. The balancing strategy improves the learning capability of the anomaly detection models and contributes to more reliable and unbiased detection performance.

While feature extraction can be effective, it increases the risk of overfitting, particularly in high-dimensional or limited datasets. In the context of Network Intrusion Detection systems for IoT security, several feature selection techniques have been widely implemented, including Gini impurity, Chi-square test, Information Gain, Mutual Information, and Feature Correlation [14].

3.5. Proposed Hybrid Feature Selection

Feature selection methods are used to achieve the most relevant features for intrusion detection system by eliminating irrelevant and redundant features from high-dimensional IoT network traffic data [15]. In the study, a hybrid feature selection approach is proposed that integrates filter-based method and embedded method. Figure 2 shows the proposed hybrid feature selection framework designed to identify the most discriminative features for anomaly detection in IoT network intrusion detection systems.

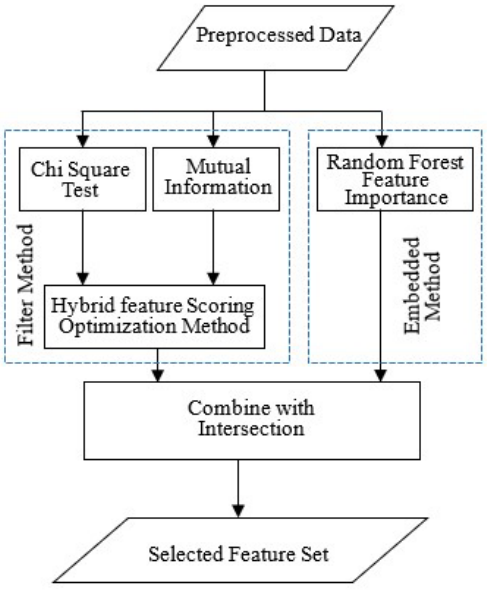

Figure 2: Proposed hybrid feature selection method (author designed)

The methodology integrates filter-based and embedded feature selection techniques in a unified framework. In the first stage, two filter-based feature selection techniques, namely chi-square and mutual information, are applied independently. The chi-square method evaluates the statistical dependency between individual features and class labels to identify features with strong categorical associations. Mutual information measures the amount of shared information between features and target classes to capture of both linear and non-linear relationships. The scores obtained from the two methods are subsequently fused through a hybrid feature scoring mechanism, which reduces feature redundancy and enhances the relevance of the selected subset.

In parallel, an embedded feature selection approach based on Random Forest feature importance is employed. It considers feature relevance during model training and captures complex feature interactions inherent in IoT traffic data. Finally, the outputs of the hybrid filter-based scoring and the embedded method are combined to generate an optimal feature subset. The integrated strategy improves detection performance while maintaining computational efficiency, making the framework suitable for large-scale and real-time IoT intrusion detection environments.

3.5.1. Chi-Square Test

The Chi-Square (χ^2) test is a statistical method used to assess the independence between categorical input features and the target class. The feature selection for classification tasks quantifies the degree of association between each feature and the class label. A higher Chi-Square score indicates a stronger dependency, suggesting that the feature is more relevant for classification.

Algorithm 1: Hybrid Feature Selection Method for Intrusion Detection System

Input: Dataset D with features $F = \{f_1, f_2, \ldots f_n\}$, labels Y, parameters α, β such that $\alpha+\beta=1$
Output: Selected feature subset $F_{selected}$

Step 1: Label Encoding
Encode all categorical features in F and the label vector Y into numeric format
Step 2: Drop Labels
Remove label column Y from dataset D, keeping only feature matrix
Step 3: Feature Normalization
Normalize all features in F to bring them onto a common scale using Min-Max normalization
Step 4: Compute Filter Scores
for each feature $f_i \in F$ **do**
Compute the Chi-Square score $S_{\chi 2}(f_i)$ with respect to Y
Compute Mutual Information score $S_{MI}(f_i)$ with respect to Y
end for
Step 5: Combine Filter Scores
$S_{filter(fi)} = \alpha \cdot S_{\chi 2(fi)} + \beta \cdot S_{MI(fi)}$
Step 6: Compute Embedded Scores (Random Forest Importance)
Train a Random Forest classifier using features F and labels Y
Extract feature importance scores $S_{RF(fi)} = RF_{Importance(fi)}$
Step 7: Feature Subset Selection
$F_{topFilter}$: top k features rank by S_{filter}
F_{topRF} : top k features rank by S_{RF}
$F_{selected} = F_{topFilter} \cap F_{topRF}$
return $F_{selected}$

The most informative features are selected for model training based on either a predefined threshold or a specified number of top-ranked features (k). A threshold is applied in the proposed method to retain the top 50% of features, selected based on their p-values derived from the Chi-square test of feature importance scores. The Chi-Square test is particularly effective for high-dimensional datasets, making it suitable for feature selection in complex IoT network intrusion detection systems [16]. For each feature, the test calculates the Chi-Square statistic in (1).

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \quad (1)$$

where, O_i is the observe frequency and E_i is the expected frequency for each category.

By measuring the degree of dependency between individual features and class labels, the Chi-Square method highlights feature that exhibit strong discriminatory power between normal and malicious traffic patterns. The method employs with low computational complexity and classifier independent nature. Therefore, it is suitable for large-scale IoT environments with limited processing resources. Moreover, it provides a reliable preliminary feature ranking that facilitates dimensionality reduction while preserving critical intrusion-related information.

3.5.2. Mutual Information

Mutual Information is a filter-based feature selection technique that qualifies the amount of shared information between an input feature and the target class, making it suitable for discrete and continuous variables. In the context of IoT-based intrusion detection, MI serves as an information-theoretic metric that quantifies the reduction in class label uncertainty provided by a given feature.

Features with higher MI scores are considered more informative for classification tasks. Unlike linear correlation methods, MI captures non-linear dependencies between features and labels suitable for complex datasets. In the system, MI is applied to assess the relevance of each feature concern the attack label, contributing to the hybrid selection strategy that enhances detection performance while reducing dimensionality [17]. MI is completed using the joint probability distribution P(x, y) and the marginal probability P(x), P(y) in (2).

$$MI(X;Y) = \sum_{x\in X}\sum_{y\in Y} P(x,y) log \frac{P(x,y)}{P(x)P(y)} \quad (2)$$

MI feature selection is used in the proposed system to capture the informational dependency between network traffic features and intrusion classes. Unlike traditional statistical methods, it identifies both linear and non-linear relationship in complex and dynamic IoT network traffic patterns.

3.5.3. Random Forest Feature Importance

RF is a widely used ensemble learning algorithm that not only provides strong predicted performance but also offers an embedded mechanism for feature selection [18]. The importance of feature is determined by the extent to which it decreases the impurity of nodes across the ensemble. The importance scores $I(f_i)$ of a feature f_i is computed as the total decrease in Gini impurity brought by that feature, averaged over all trees in the forest as in (3).

$$I(f_i) = \sum_{t=1}^{T} \sum_{n\in N_t(f_i)} \frac{w_n}{W} . \left[Gini(n) - \left(\frac{w_{n_L}}{w_n} Gini(n_L) + \frac{w_{n_R}}{w_n} Gini(n_R) \right) \right] \quad (3)$$

where, T is the total number of trees. $N_t(f_i)$ is the set of nodes in tree t where feature f_i is used to split, w_n is the weighted number of samples reaching node n, W is the total number of weight, w_n/W is the importance weight of that node, w_{nL} and w_{nR} are the number of samples in the left and right child nodes respectively, and W is the total number of samples in the dataset. Gini(n), Gini(n_L), and Gini(n_R) denote the Gini impurity of the parent and child nodes.

RF considers feature contributions within an ensemble of decision trees to capture complex and non-linear interactions among IoT network traffic attributes. It is beneficial for intrusion detection where attack behaviours often manifest through intricate feature dependencies. Consequently, features that significantly enhance the purity of the resulting subsets are assigned higher importance scores, indicating their stronger relevance in the classification process [19].

3.5.4. Hybrid Feature Selection

The hybrid feature selection stage integrates the outputs of Chi-square, mutual information, and RF feature importance to construct a compact and highly discriminative feature subset for IoT intrusion detection. The feature score obtained from Chi-square test and MI are combined as in (4) using a weighted scheme controlled by parameters α and β, where α + β =1.

$$S_{filter(fi)} = \alpha \cdot S_{\chi 2(fi)} + \beta \cdot S_{MI(fi)} \quad (4)$$

where, $S_{filter(fi)}$ denotes the importance score for feature fi, $S_{\chi 2(fi)}$ for Chi-square, and $S_{MI(fi)}$ is for MI.

The parameter α controls the contribution of the Chi-square score, while β governs the contribution of the Mutual Information (MI) score in the hybrid ranking computation. The analysis is to determine how sensitive the final selected feature subset and the subsequent classification performance are to variations in the weighting configuration in (5) to (11).

To ensure a theoretically grounded weighting scheme, the constraint $\alpha + \beta = 1$ can be interpreted from a minimum-risk estimation perspective. Let the true relevance of a feature X be denoted by $R(X)$. The Chi-square and Mutual Information (MI) scores can be viewed as noisy estimators of this true relevance:

$$R_\chi = R(X) + \varepsilon_\chi, R_{\mathrm{MI}} = R(X) + \varepsilon_{\mathrm{MI}} \quad (5)$$

where the estimation errors have variances

$$\mathrm{Var}(\varepsilon_\chi) = \sigma_\chi^2, \mathrm{Var}(\varepsilon_{\mathrm{MI}}) = \sigma_{\mathrm{MI}}^2 \quad (6)$$

The hybrid scoring mechanism forms a weighted estimator:

$$\hat{R} = \alpha R_\chi + \beta R_{\mathrm{MI}}, \text{ with } \alpha + \beta = 1. \quad (7)$$

Assuming the estimation errors are unbiased and independent, the variance of the combined estimator becomes

$$\mathrm{Var}(\hat{R}) = \alpha^2 \sigma_\chi^2 + \beta^2 \sigma_{\mathrm{MI}}^2 \quad (8)$$

Minimizing this variance yields the classical optimal weights:

$$\alpha^* = \frac{\sigma_{\mathrm{MI}}^2}{\sigma_\chi^2 + \sigma_{\mathrm{MI}}^2}, \beta^* = \frac{\sigma_\chi^2}{\sigma_\chi^2 + \sigma_{\mathrm{MI}}^2} \quad (9)$$

The result provides important insight for the proposed hybrid scheme. When the MI estimator is more reliable (i.e., $\sigma_{\mathrm{MI}}^2 < \sigma_\chi^2$), the optimal solution naturally assigns a larger weight to MI ($\beta > \alpha$). Therefore, the empirical configuration used in the proposed method ($\alpha = 0.3, \beta = 0.7$) is mathematically justified when MI provides more stable relevance estimates for IoT traffic features.

From an error analysis viewpoint, feature ranking using a single method suffers from the classical bias–variance trade-off:

$$E_{\mathrm{single}} = \mathrm{Bias}^2 + \mathrm{Variance} \quad (10)$$

The proposed hybrid approach reduces the variance component through weighted averaging of partially independent estimators, leading to

$$E_{\mathrm{hybrid}} < E_{\mathrm{single}} \quad (11)$$

Consequently, the hybrid feature selection mechanism produces a more robust and discriminative feature subset, which explains the observed improvement in downstream classification accuracy for IoT intrusion detection.

The sensitivity test was conducted using five different parameter configurations. The scores obtained from both filter-based methods are normalized and subsequently fused to generate a unified feature ranking. The fusion mechanism reduces redundancy while preserving features that consistently demonstrate high relevance across multiple evaluation criteria [20]. RF feature importance is employed as an embedded refinement step to further evaluate the selected features within a learning-based context. It captures nonlinear feature interactions and validates the contribution of each feature to classification performance. The hybrid strategy combines computational efficiency with learning-aware selection to improve detection accuracy and reduces dimension for scalable IoT intrusion detection systems.

3.6. Classification

The selected optimal feature subset is utilized to identify anomalous and malicious activities in IoT network traffic. To comprehensively evaluate the proposed hybrid feature selection framework, both binary classification and multiclass classification scenarios are considered. Binary classification is employed to distinguish normal traffic from attack traffic, while multiclass classification is used to identify specific attack categories present in the IoT environment. Five machine learning classifiers are adopted due to the diverse learning characteristics and proven effectiveness in intrusion detection tasks. Random forest classifier [21] constructs a collection of decision trees during the training process and combines the outputs to improve accuracy as in (12).

$$\hat{Y} = Mode\left(\{T_b(X)\}_{b=1}^{B}\right) \quad (12)$$

Where, $\hat{Y}$ is the final predicted class. B is the total number of decision trees in the forest. $T_b(X)$ is the prediction of an individual tree b for the input data X. Mode refers to the Majority vote among all individual tree prediction.

RF classifier reduces overfitting and variance by aggregating the prediction of multiple independently trained decision trees. Each tree is built using a bootstrap sample drawn randomly from the training dataset and a random subset of feature is selected at each node to determine the optimal split [22]. This randomness enhances model generalization and makes effective for complex and high-dimensional IoT intrusion detection data.

3.6.1. Decision Tree classifier

Decision tree is a supervised learning algorithm that models the decision-making process through a tree-like structure composed of internal decision nodes, branches, and leaf nodes. Each internal node represents a test on a feature, branches correspond to the outcomes of the test, and leaf nodes denote class labels [23]. It is used in intrusion detection system due to the simplicity, interpretability, and ability to handle both numerical and categorical features without requiring extensive data preprocessing.

During training, the classifier recursively partitions the input dataset by selecting features that best separate the data into distinct classes. The selection of the optimal feature at each node is based on an impurity measure, commonly Information Gain, which is derived from entropy. The entropy of a dataset S is defined in (13).

$$Gini(S) = 1 - \sum_{i=1}^{c} p_i^2 \quad (13)$$

where, p_i is the proportion (probability) of data points belonging to class *i* in the dataset *S*.

3.6.2. K-nearest Neighbor classifier

K-nearest neighbour is a non-parametric, instance-based learning algorithm that classifies an input sample based on the similarity between the sample and its nearest neighbours in the feature space. It does not require an explicit training phase and performs classification at the time of prediction [24][25]. The classification algorithm computes the distance between input instance and all training samples and identifies the k closest neighbors. Euclidean distance is defined in (14).

$$d(x, x_i) = \sqrt{\sum_{j=1}^{n}(x_j - x_{ij})^2} \quad (14)$$

where x_j and x_{ij} represent the values of the j^{th} feature of the test instance and the i^{th} training instance, respectively, and n denotes the number of features. Due to the reliance on distance computation, the performance is highly influenced by feature scaling and dimensionality. The proposed hybrid feature selection method significantly enhances the efficiency and accuracy of the classifier for both binary and multiclass IoT intrusion detection tasks.

3.6.3. Support Vector Machine classifier

Support vector machine is a supervised learning algorithm that is widely used for intrusion detection due to its strong classification in high-dimensional feature spaces [24]. It constructs an optimal separating hyperplane that maximizes the margin between different classes to improve generalization performance [26]. It is suitable for both binary and multiclass IoT intrusion detection tasks. For a linear SVM, the decision function f(x) is a simple linear equation that defines the decision boundary as in (15).

$$f(\mathrm{x}) = \mathrm{w}^T\mathrm{x} + b \quad (15)$$

where, x is the input feature vector of the data point to be classified. w is the weighted vector which is perpendicular to the hyperplane and points toward the positive class b. The symbol b is the bias which shifts the hyperplane away from the origin. The classification rule is:

If $f(\mathbf{x}) \geq 0$, classify as the positive class (+1).
If $f(\mathbf{x}) < 0$, classify as the negative class (-1).

3.6.4. Naïve Bayes classifier

The Naïve Bayes classifier [27] is a probabilistic supervised learning algorithm based on Bayes' theorem and the assumption of conditional independence among features given the class label as in (16).

$$P(y \mid x_1, \ldots, x_n) = \frac{P(x_1,\ldots,x_n|y)P(y)}{P(x_1,\ldots,x_n)} \quad (16)$$

where,

$P(y \mid X)$**:** Probability of class y given feature vector X

$P(X \mid y)$**:** Probability of features given class y.

$P(y)$**:** Initial probability of class y.

$P(X)$**:** Marginal likelihood, often ignored during prediction as it is constant for all classes.

It is used in IoT network environments for low computational complexity, scalability, and handling high-dimensional data.

4. Datasets

To evaluate the performance of the proposed system using hybrid feature selection method, three IoT intrusion detection datasets are employed in the study: ACT-IoT2023, BoT-IoT, and TON-IoT datasets. The datasets represent diverse IoT environments and attack scenarios in terms of network traffic, attack types, and feature distributions.

4.1. ACI-IoT Dataset

The ACI-IoT 2023 dataset consists of real-world IoT network traffic collected from a variety of smart devices and communication protocols. It includes both benign and malicious activities, covering multiple categories such as denial-of-service (DoS), data theft, probing, and botnet attacks as shown in Table 1. Each record in the dataset contains rich flow-based, statistical, and protocol-specific features, which enable effective representation of IoT network behaviour [28].

The attack types are Benign, Port scan, ICMP Flood, Ping Sweet, DNS Flood, Vulnerability Scan, OS Scan, Slowloris, SYN Flood, Dictionary attack, UDP flood, ARP Spoofing. This dataset is particularly suitable for evaluating the adaptability of intrusion detection models to contemporary IoT network environments. The numbers of instance are presented in Table 1.

Table 1: Statistics of ACI-IoT dataset Table

Class	No. of instances
Benign	329295
Port scan	441282
ICMP Flood	225234
Ping Sweet	71928
DNS Flood	46935
Vulnerability Scan	39537
OS Scan	37524
Slowloris	18643
SYN Flood	13857
Dictionary attack	6380
UDP flood	791
ARP Spoofing	5

4.1. BoT-IoT Dataset

The BoT-IoT dataset, developed by the Cyber Range Lab at UNSW Canberra, is a large-scale dataset that emulates realistic IoT network traffic, incorporating both legitimate and attack flows [29]. The includes normal and six attacks: DDoS, Dos, Reconnaissance, Theft, Backdoor, Injection. The dataset is widely used for benchmarking intrusion detection algorithms, as it provides a balance between data volume, feature richness, and realistic attack patterns. However, due to its synthetic nature, it sometimes exhibits class imbalance issues that must be addressed during model training and evaluation. The numbers of instance are described in Table 2.

Table 2: Statistics of BoT-IoT dataset Table

Class	No. of instances
Normal	664
DDoS	481611
DoS	481611
Reconnaissance	95289
Theft	386322
Backdoor	424043
Injection	538515

4.1. TON-IoT Dataset

The TON-IoT dataset represents a comprehensive collection of telemetry, network, and long data from various IoT and Industrial IoT (IIoT) environments [30]. It contains data streams from sensors, edge devices, and network layers, enabling both network-level and device-level intrusion analysis. The dataset multi-source nature allows the assessment of the proposed method's robustness and scalability across heterogenous IoT data domains. This dataset includes 10 attacks: Benign, Generic, Exploits, Fuzzers, Reconnaissance, DoS, Backdoors, Analysis, Shellcode, Worms. The number of instances in each attack are presented in Table 3. The number of total instances and original features in the three datasets are described in Table 4.

Table 3: Statistics of TON-IoT dataset Table

Class	No. of instances
Benign	677786
Generic	7522
Exploits	5409
Fuzzers	5051
Reconnaissance	1759
DoS	1167
Backdoors	534
Analysis	526
Shellcode	223
Worms	24

Table 4: Number of instances and original features

Dataset	No. of instances	No. of features
ACI-IoT	1231411	85
BoT-IoT	2408055	46
TON-IoT	700001	45

5. Experimental Results

The proposed system with hybrid feature selection method is evaluated on three datasets. The system is tested under both binary and multiclass classification scenarios to evaluate the performance of the system. Comprehensive experiments are conducted using five machine learning classifiers to analyse the impact of the proposed feature selection method. All experiments are implemented using Python-based machine learning libraries on a personal computer equipped with an Intel Core i5 processor operating at 3.2 GHz, 8 GB of RAM. The system runs on a 64-bit Window 11 operating system, which provides a stable environment for machine learning model training and evaluation.

5.1. Feature Selection Results

In the proposed ensemble feature selection method, the weighted values are set as $\alpha = 0.3$ and $\beta = 0.7$. The resulting composite scores, obtained from the weighted combination of Chi-square and Mutual Information, and then integrated with the feature importance scores derived from the Random Forest algorithm using intersection-based strategy. Each method selects (top k) 50% of original features. Only BoT-IoT dataset will be used SMOTE method, because it has imbalanced class distribution and a large fraction of various attack types. The numbers of selected feature for each dataset are described in Table 5.

In the study, the proposed hybrid feature selection method is evaluated by comparing it with individual feature selection techniques such as Chi-square test, MI and RF importance. The number of features is identical for each dataset, as the top 50% of features were consistently selected based on their respective importance scores. This uniform threshold ensured a fair comparison to identifying relevant features.

Table 5: Numbers of selected feature on each datasets

Dataset	Number of Features		
	Original	Chi-Square/ MI/ RF	Proposed Hybrid Feature Selection
ACI-IoT	85	34	28
BoT-IoT	46	21	13
TON-IoT	45	22	13

However, when reducing the selection threshold to 35%, the number of selected features significantly decreased. While this reduction aimed to further eliminate less relevant features, it also resulted in too few common features remaining across all three methods. Consequently, the intersection set became too small, limiting the effectiveness of the hybrid feature selection process. To maintain a balance between dimensionality reduction and feature relevance, the 50% threshold was retained. It provided a sufficient number of intersecting features for the proposed hybrid approach while preserving important characteristics from each scoring method.

The features selected by each method are then used to classify the data using five supervised machine learning algorithms: RF, k-NN, DT, SVM, NB classifiers.

5.2. Evaluation metrics

Performance evaluation is carried out using standard metrics, including accuracy, precision, recall, and F1-score. Accuracy measures the proportion of correctly classified instances as in (17). Precision quantifies the proportion of correctly predicted positive instances among all predicted positives as in (18).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (17)$$

where, TP is true positive, TN is true negative, FP is false positive, FN is false negative values.

$$Precision = \frac{TP}{TP+FP} \quad (18)$$

Recall, known as sensitivity, reflects the proportion of correctly predicted positive instances among all actual positives

as in (19). F1-score is calculated as the harmonic mean of precision and recall, provides a balanced measure of classification effectiveness as in (20).

$$Recall = \frac{TP}{TP+FN} \tag{19}$$

$$F1 - Score = \frac{2.Precision\ .Recall}{Precision\ +Recall} \tag{20}$$

5.3. Data Splitting

In the study, the dataset was partitioned into three subsets to ensure a fair and robust evaluation of the proposed model. Specifically, 60% of the data was allocated for training to enable the model to learn the underlying patterns and relationships within the features, 20% was reserved for validation to fine-tune the model parameters and prevent overfitting, and the remaining 20% was used for testing to objectively assess the model generalization performance on unseen data. This stratified data splitting strategy ensures that all subsets maintain a representative class distribution, thereby enhancing the reliability and reproducibility of the experimental results.

5.4. Sensitivity Analysis

The top-ranked features were selected and intersected with the RF feature importance scores to form the final feature subset. The results show that the hybrid model achieves consistent performance across different weight combinations, with only slight variations in classification metrics. The configuration (α = 0.3 and β = 0.7) achieves the highest overall accuracy across all datasets. This indicating that contributes more effectively to distinguishing relevant features in the context of IoT network intrusion detection. The sensitivity analysis results present in Figure. 3, Figure. 4, and Figure. 5.

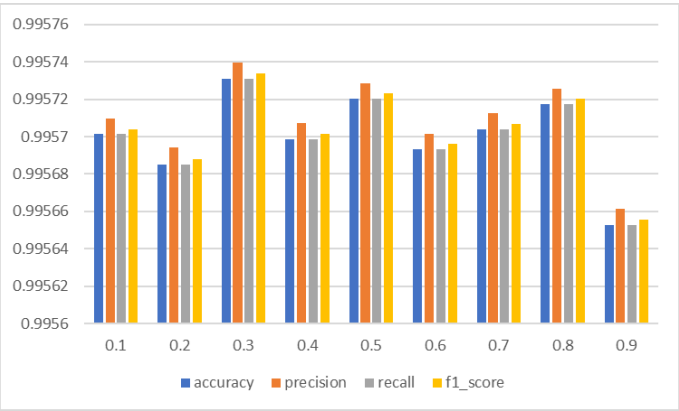

Figure 3. Sensitivity analysis of the hybrid feature selection method on ACI-IoT dataset

For all three datasets, the performance metrics remain relatively consistent across the range of tested α values, indicating that the method is not overly sensitive to minor changes in the weighting parameters. Assigning a slightly higher weight to the MI score enhances the ability of the hybrid method to capture nonlinear relationships between features and class labels. These findings confirm that the proposed hybrid approach achieves a balanced integration of statistical relevance and dependency strength.

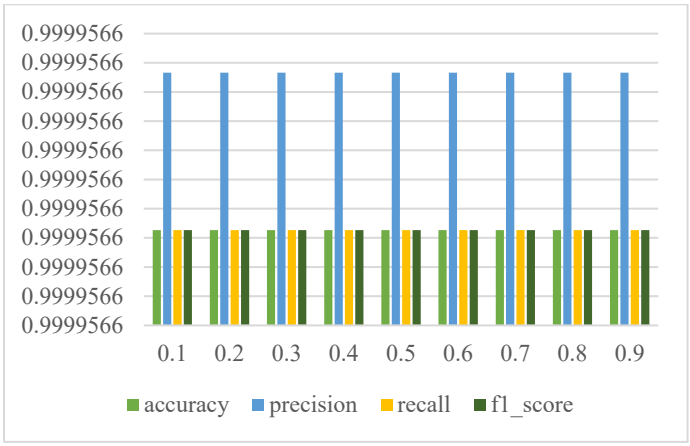

Figure 4. Sensitivity analysis of the hybrid feature selection method on BoT-IoT dataset

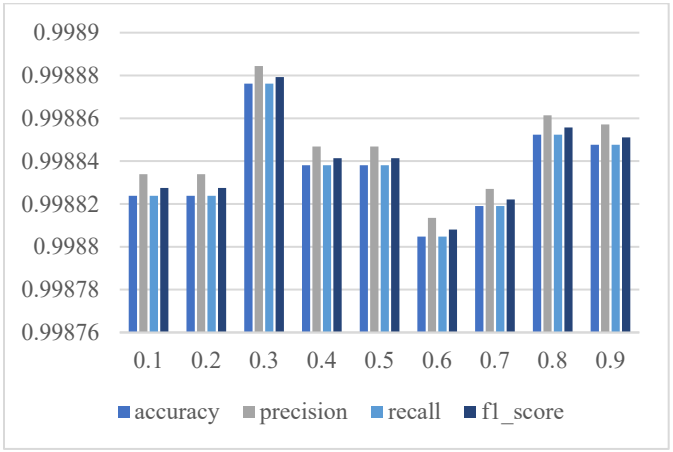

Figure 5. Sensitivity analysis of the hybrid feature selection method on TON-IoT dataset

5.5. Classification Results

Table 6 summarizes the classification performance of the proposed intrusion detection system on the ACT-IoT dataset under both binary and multiclass classification settings. The RF classifier achieves the highest performance across all evaluation metrics, obtaining an accuracy of 99.98% for both binary and multiclass tasks. Decision Tree and K-NN classifier also deliver highly competitive results, with accuracy values close to RF that confirms the effectiveness of the selected hybrid features. The SVM classifier performs well in binary classification but exhibits a performance drop in the multiclass scenario, indicating challenges in modelling complex class separations. The NB classifier shows high recall but relatively lower precision in binary classification, reflecting a higher false alarm rate, while its multiclass performance improves significantly.

Table 7 presents the performance on BoT-IoT dataset. The RF classifier achieves the high performance with an accuracy of 99.99% for binary and 99.98% for multiclass classification, along with consistently high precision, recall, and F1-score values. DT

and K-NN classifiers also demonstrate outstanding performance, closely matching RF with minimal performance degradation, indicating the effectiveness of the selected features in capturing attack patterns. In contrast, SVM shows comparatively lower performance, particularly in the multiclass scenario, suggesting limitations in handling complex class boundaries. The NB classifier exhibits the weakest performance, characterized by high recall but low precision, leading to increased false alarms.

In Table 8, The RF classifier achieves the highest overall performance with highest accuracy and balanced precision-recall values in both classification settings which indicates the strong capability to model complex IoT traffic pattern. DT and K-NN classifier show competitive performance with accuracy values close to RF. The SVM delivers moderate results with reduced precision in binary classification, reflecting a higher misclassification rate for certain attack types. The NB classifier exhibits very high recall but relatively low precision in binary classification task which leads to increased false alarms.

Accordingly, to results, RF achieves the highest performance than other classifiers across all datasets. The proposed hybrid feature selection approach enhances the detection accuracy and reliability across diverse IoT network environments. The confusion metrics using proposed hybrid feature selection and RF classifier are shown in Figure. 6, Figure. 7, and Figure. 8 for binary classification. The confusion metrices of multiclass classification are shown in Figure. 9, Figure. 10, Figure. 11 for three datasets respectively.

5.6. Performance Comparison

In Table 9, the performance comparison of original features, Chi-square selected features, features with MI, feature with RF and proposed hybrid feature selection method on three datasets. The results indicate that individual feature selection methods provide marginal performance improvements over original feature set across all datasets. The proposed hybrid feature selection method either matched or slightly improved detection performance while reducing execution time. Therefore, the hybrid method stability across diverse datasets and data imbalance conditions confirms the robustness and generalizability.

Table 6 Classification result on ACI IoT Dataset

	Binary Classification				Multi-class Classification			
Method	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
RF	0.9993	0.9995	0.9996	0.9995	0.9993	0.9993	0.9993	0.9993
DT	0.9989	0.9992	0.9993	0.9992	0.9989	0.9989	0.9989	0.9989
k-NN	0.9989	0.9992	0.9993	0.9992	0.9989	0.9989	0.9989	0.9989
SVM	0.9944	0.9987	0.9995	0.9988	0.8736	0.9091	0.8736	0.8844
NB	0.8379	0.8233	0.9915	0.8996	0.9333	0.9611	0.9383	0.9455

Table 7 Classification result on BoT IoT Dataset

	Binary Classification				Multi-class Classification			
Method	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
RF	0.9999	1.000	0.9998	0.9999	0.9998	0.9998	0.9998	0.9998
DT	0.9999	0.9999	0.9998	0.9999	0.9996	0.9996	0.9996	0.9996
k-NN	0.9999	0.9999	0.9998	0.9999	0.9996	0.9996	0.9996	0.9996
SVM	0.9685	0.9756	0.9610	0.9683	0.8962	0.9031	0.8962	0.8972
NB	0.6360	0.5800	0.9861	0.7304	0.7473	0.8508	0.7473	0.7414

Table 8 Classification result on BoT IoT Dataset

	Binary Classification				Multi-class Classification			
Method	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score
RF	09990	0.9799	0.9876	0.9834	0.9901	0.9889	0.9901	0.9892
DT	0.9983	0.9762	0.9734	0.9734	0.9891	0.9882	0.9891	0.9885
k-NN	0.9983	0.9762	0.9734	0.9730	0.9891	0.9882	0.9891	0.9885
SVM	0.9946	0.8675	0.9811	0.9208	0.9584	0.9722	0.9584	0.9633
NB	0.9870	0.7094	0.9998	0.8229	0.9618	0.9766	0.9618	0.9684

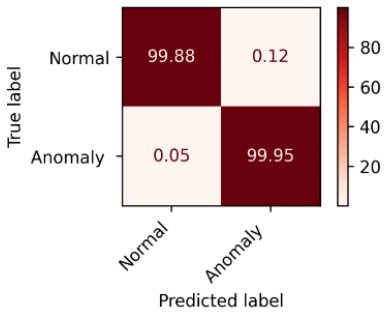

Figure 6. Confusion matrix of binary classification on ACI-IoT dataset

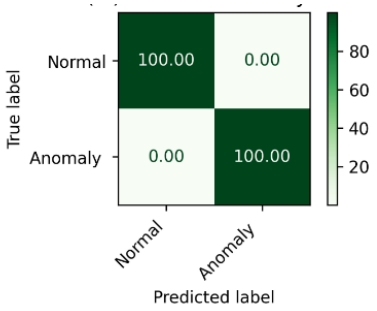

Figure7. Confusion matrix of binary classification on BoT-IoT dataset

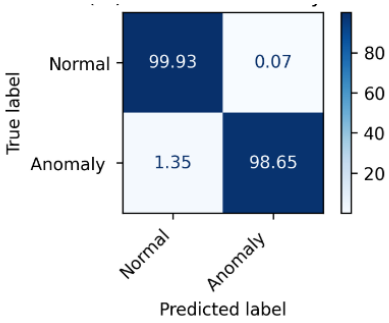

Figure 8. Confusion matrix of binary classification On TON-IoT dataset

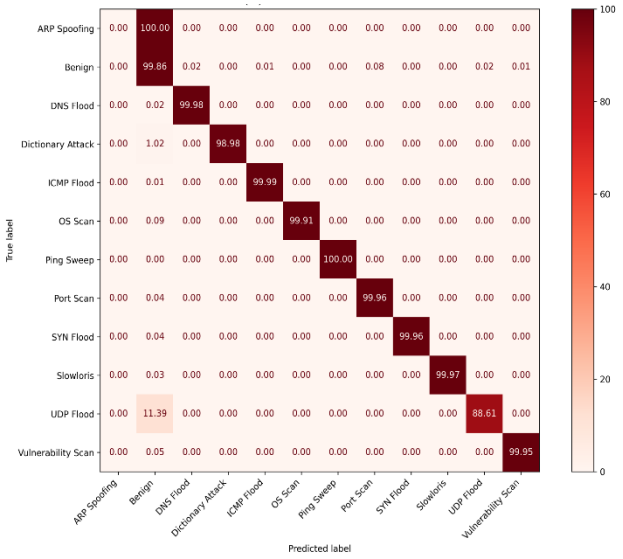

Figure 9. Confusion matrix of multiclass classification on ACI-IoT dataset

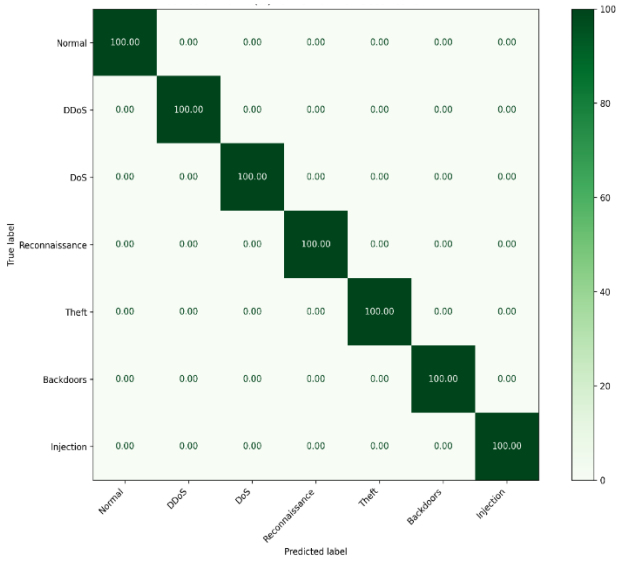

Figure 10. Confusion matrix of multiclass classification on BoT-IoT dataset

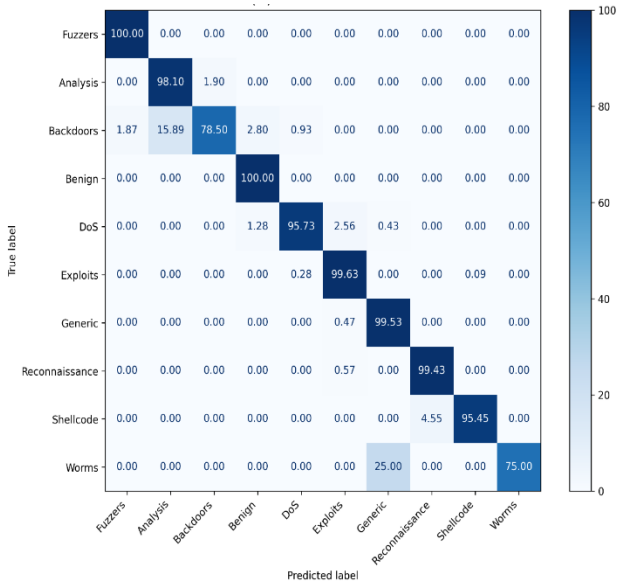

Figure 11. Confusion matrix of multiclass classification on TON-IoT dataset

Table 9: Comparison of classification accuracy on three datasets using RF classifier

Features	ACI-IoT	BoT-IoT	TON-IoT
Original	0.9987	0.9999	0.9982
Chi-square	0.9988	0.9999	0.9982
MI	0.9992	0.9999	0.9963
RF	0.9993	0.9999	0.9966
Proposed	0.9993	0.9999	0.9990

Table 10 presents the comparison of proposed system and state-of-the-art researches. The results indicates that the hybrid approach reduces number of features while retaining the high performance.

Table 10. Comparison of performance with Previous Studies

Reference	Methods	Dataset	Scenario	Accuracy	Precision	Recall	F1-Score
[6]	Deep Learning	TON-IoT	binary	0.9987	0.9987	0.9987	0.9987
			multiclass	0.9949	0.9949	0.9949	0.9949
[8]	Hybrid feature selection, Multistage classification	TON-IoT	multiclass	0.9883	0.9856	09876	0.9865
		BoT-IoT	multiclass	0.9860	0.9850	0.9863	09894
[9]	MI+SFS, Random Forest	BoT-IoT	binary	0.9999	0.9958	0.9947	0.9953
[10]	RNN (MLP)	BoT-IoT	multiclass	0.8400	0.8400	0.8400	0.8200
Proposed	Hybrid feature selection, RF	TON-IoT	binary	09990	0.9799	0.9876	0.9837
		TON-IoT	multiclass	0.9998	0.9998	0.9998	0.9998
		BoT-IoT	binary	0.9999	1.0000	0.9998	0.9999
		BoT-IoT	multiclass	0.9998	0.9998	0.9998	0.9998

6. Conclusion

The proposed hybrid feature selection method integrates filter techniques with an embedded method based on Random Forest importance scores through a hybrid feature scoring optimization strategy. The experimental results demonstrate that the approach improves anomaly detection performance while effectively reducing feature dimensionality across five supervised machine learning classifiers. Among the classifiers, RF attains the highest performance with accuracy of 99.93%, 99.99%, 0.99.93% on ACI-IoT2023, BoT-IoT, and TON-IoT datasets respectively. In future work, the proposed feature selection approach will be extended to deep learning models to explore their potential in capturing complex spatial and temporal patterns in IoT traffic. Additionally, the model will be optimized for deployment on resource-constrained IoT devices to facilitate real-time and energy-efficient anomaly detection at the network edge with minimal computational overload.

Conflict of Interest

The authors declare no conflict of interest.

Acknowledgment

For the conceptualization, MSSMoe; methodology, MSSMoe; analysis, MSSMoe; data collection, MSSMoe; investigation and validation, MSSMoe and WMOo, and writing paper and editing, MSSMoe; and visualization, WMOo and MSSMoe; Supervision WMOo. All authors have read and edit to the manuscript.

The authors would like to express their sincere gratitude to the researchers and institutions who developed and made publicly available the ACI-IoT2023, BoT-IoT, TON-IoT datasets. The authors appreciate the efforts of the dataset creators in supporting the research community through open and accessible data resources.

References

[1] T.M. Ghazal, M.K. Hasan, M.T. Alshurideh, H.M. Alzoubi, M. Ahmad, S.S. Akbar, B. Al Kurdi, I.A. Akour, "IoT for Smart Cities: Machine Learning Approaches in Smart Healthcare—A Review," Future Internet, **13**(8), 218, 2021, doi:10.3390/fi13080218..

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, M. Zorzi, "Internet of Things for Smart Cities," IEEE Internet of Things Journal, **1**(1), 22–32, 2014, doi:10.1109/JIOT.2014.2306328.

[3] E. Gyamfi, A. Jurcut, "Intrusion detection in internet of things systems: A review on design approaches leveraging multi-access edge computing, machine learning, and datasets," Sensors, **22**(10), 1–33, 2022, doi.org/10.3390/s22103744

[4] H. Wang, T.M. Khoshgoftaar, K. Gao, "A comparative study of filter-based feature ranking techniques," in IEEE International Conference on Information Reuse and Integration, 43–48, 2010, doi.org/10.1109/IRI.2010.5558913

[5] M.A.M. Hasan, M. Nasser, S. Ahmad, K.I. Molla, "Feature selection for intrusion detection using random forest," Journal of Information Security, **7**, 129–140, 2016, doi:10.4236/jis.2016.73009

[6] D.S. Ahmed, "Anomaly-based network intrusion detection system for IoT using deep learning model," Scientific Research Journal of Engineering and Computer Science, 3(5), 16–23, 2023, doi: 10.47310/srjecs.2023.v03i02.010

.[7] G. Guo, X. Pan, H. Liu, F. Li, L. Pei, K. Hu, "An IoT intrusion detection system based on TON-IoT network dataset," in IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), 333–338, 2023, doi.org/10.1109/CCWC57344.2023.10099144

[8] G. Logeswari, J.D. Roselind, K. Tamilarasi, V. Nivethitha, "A comprehensive approach to intrusion detection in IoT environments using hybrid feature selection and multi-stage classification techniques," IEEE Access, **13**, 24970–24987, 2025.

[9] A.G. Ayad, N.A. Sakr, N.A. Hikal, "A hybrid feature selection model for anomaly-based intrusion detection in IoT networks," in International Telecommunications Conference (ITC-Egypt), 1–7, 2024.

[10] F.A. Alotaibi, S. Mishra, "Cyber security intrusion detection and bot data collection using deep learning in the IoT," International Journal of Advanced Computer Science and Applications, **15**(3), 421–432, 2024.

[11] I. Kerrakchou, A.A. El Hassan, S. Chadli, M. Emharraf, M. Saber, "Selection of efficient machine learning algorithm on BoT-IoT dataset for intrusion detection in internet of things networks," Indonesian Journal of Electrical Engineering and Computer Science, **31**(3), 1784–1793, 2023, doi:10.11591/ijeecs.v31.i3.pp1784-1793

[12] J. Han, J. Pei, M. Kamber, Data Mining: Concepts and Techniques, 3rd ed., Elsevier, 2011, ISBN 9780123814791.

[13] N. V Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, **16**, 321–357, 2002, doi:10.1613/jair.953.

[14] J. Li, M.S. Othman, H. Chen, L.M.M. Yusuf, "Optimizing IoT intrusion detection system: Feature selection versus feature extraction in machine learning," Journal of Big Data, **11**(1), 36, 2024, doi.org/10.1186/s40537-024-00892-y.

[15] S. Walling, S. Lodh, "Network intrusion detection system for IoT security using machine learning and statistical based hybrid feature selection," Security and Privacy, **7**(6), 2024, doi:10.1002/spy2.429

[16] P.N. Tan, M. Steinbach, V. Kumar, Introduction to Data Mining, Pearson Education India, 2016.

[17] N. Hoque, D.K. Bhattacharyya, J.K. Kalita, "MIFS-ND: A mutual information-based feature selection method," Expert Systems with Applications, **41**(14), 6371–6385, 2014, doi:10.1016/j.eswa.2014.04.019.

[18] L. Breiman, "Random forests," Machine Learning, **45**(1), 5–32, 2001..

[19] X. Yuan, S. Liu, W. Feng, G. Dauphin, "Feature importance ranking of random forest-based end-to-end learning algorithm," Remote Sensing, **15**(21), 5203, 2023, doi: 10.3390/rs15215203.

[20] P. Rani, R. Kumar, A. Jain, "A hybrid approach for feature selection based on correlation feature selection and genetic algorithm," International Journal of Software Innovation, **10**(1), 1–17, 2022, doi: 10.4018/IJISMD.2021040102

[21] V. Kantharaju, others, "Machine learning based intrusion detection framework for detecting security attacks in internet of things," Scientific Reports, **14**(1), 30275, 2024, doi:10.1038/s41598-024-81535-3

[22] A. Liaw, M. Wiener, "Classification and regression by random forest," R News, **2**(3), 18–22, 2002.

[23] J.R. Quinlan, "Induction of decision trees," Machine Learning, **1**(1), 81–106, 1986.

[24] T.M. Mitchell, Machine Learning, McGraw-Hill, 1997.

[25] T. Cover, P. Hart, "Nearest neighbor pattern classification," IEEE Transactions on Information Theory, **13**(1), 21–27, 1967, doi:10.1109/TIT.1967.1053964.

[26] C. Cortes, V. Vapnik, "Support-vector networks," Machine Learning, **20**, 273–297, 1995, doi: 10.1007/BF00994018

[27] C.J. Vargas, others, "Intrusion detection in Internet of Things systems: A feature extraction with Naive Bayes classifier approach," Journal of Machine and Computing, **4**(1), 2024, doi: 10.53759/7669/jmc202404003.

[28] Australian Centre for Internet of Things Security Research, ACI-IoT 2023 Dataset, 2023.

[29] N. Moustafa, J. Slay, The BoT-IoT dataset: A comprehensive benchmark for IoT network intrusion detection, 2020.

[30] N. Moustafa, others, TON-IoT: A multi-source dataset for evaluating federated and centralized intrusion detection systems in IoT, 2021.