electronics

MDPI

Article

Advanced Cybersecurity Framework for Detecting Fake Data Using Optimized Feature Selection and Stacked Ensemble Learning

Abrar M. Alajlan

Self-Development Skills Department, King Saud University, Riyadh 11451, Saudi Arabia; aalajlan1@ksu.edu.sa

Abstract

As smart cities continue to generate vast quantities of data, data integrity is increasingly threatened by instances of fraud. Anomalous or fake data deteriorate the process and have impacts on decision-making systems and predictive analytics. Hence, an effective and intelligent fake data detection model was designed by combining an advanced feature selection method with a robust ensemble classification framework. Initially, the raw data are eliminated by performing normalization, feature transformation, and noise filtering that enhances the reliability of the model. The dimensionality issues are mitigated by eliminating redundant features via the proposed Elite Tuning Strategy-Enhanced Polar Bear Optimization algorithm. It simulates the hunting behavior of polar bears, balancing exploration and exploitation features. The proposed Stacking Ensemble-based Random AdaBoost Quadratic Discriminant model leverages the merits of diverse base learners, including AdaBoost, Quadratic Discriminant Analysis, and Random Forest, that classify the feature subset and the integration of prediction processes with a meta-feature vector-processed meta-classifier such as a multilayer perceptron or logistic regression model that predicts the final outcome. This hierarchical architecture validates resilience against noise and improves generalization and prediction accuracy. Thus, the experimental results show that the proposed method outperforms existing approaches in terms of accuracy, precision, and latency, yielding values of 98.78%, 98.75%, and 16 ms, respectively, using the UNSW-NB15 dataset.

Keywords: IoT ecosystems; fake data detection; elite tuning strategy-enhanced polar bear optimization algorithm; AdaBoost; quadratic discriminant analysis; random forest meta-classifier

Academic Editor: Cheonshik Kim

Received: 24 June 2025
Revised: 7 August 2025
Accepted: 8 August 2025
Published: 18 August 2025

Citation: Alajlan, A.M. Advanced Cybersecurity Framework for Detecting Fake Data Using Optimized Feature Selection and Stacked Ensemble Learning. *Electronics* **2025**, *14*, 3275. https://doi.org/10.3390/electronics14163275

1. Introduction

Smart cities provide seamless integration of digital technologies to ensure quality of life and sustainability and enable real-time decision-making [1]. They rely on a continuous flow of data gathered from sensors, monitoring systems, user devices, and other digital platforms [2]. The elimination of fake or malicious data obtained in the network is a challenging task in smart cities, which are compromised by sensors, hostile attacks, or incorrect human input [3]. False data misleads automated systems responsible for traffic control, energy distribution, healthcare services, and emergency response, leading to catastrophic consequences. Thus, identifying such data ensures the reliability and accuracy of data in smart cities, maintaining the trust and functionality of smart city ecosystems [4]. The diversity and energetic nature of data sources in smart cities enable the effective

detection of complex fake data [5]. This task involves the use of an extensive communication protocol, data formats, and device capabilities that overcome the challenges of conventional data validation algorithms [6]. Furthermore, fake data generate original patterns and make effective anomaly detection methods ineffective.

Data integrity refers to the accuracy, consistency, and trustworthiness of data throughout its lifecycle, from generation to processing and storage. For instance, incorrect traffic data can mislead automated control systems, while false health sensor data may endanger lives. Ensuring high data integrity is essential for maintaining the reliability of real-time decision-making systems [7]. This enables accurate analytics and upholds public trust in smart infrastructure. In smart city ecosystems, data is continuously collected from a vast array of sources. Any compromises in data integrity, whether due to malicious tampering, sensor failure, or transmission errors, have severe consequences. Therefore, detecting and limiting fake or anomalous data at an early stage have become a critical cybersecurity requirement in modern urban environments.

Thus, advanced approaches capable of identifying complex patterns and subtle anomalies are required [8]. Numerous existing models investigate fake data detection using statistical and cognitive approaches. However, traditional models suffer from high computational complexity, poor scalability, an inability to handle imbalanced datasets, or a lack of adaptability [9]. In many cases, they fail to generalize the processed data in real-world situations and detect fake data. Thus, an effective fake data detection model that is scalable, adaptive, and robust against emerging attack strategies has been developed [10]. It distinguishes real and fake data inputs accurately in smart city applications concerning the validation of the efficiency of user behavior, data distribution, and contextual features [11].

This work focuses primarily on the detection of anomalous and fake data packets above the network layer (i.e., the network, transport, and application layers of a smart city). It is necessary to acknowledge that significant security concerns exist in the lower layers of the communication stack. Meanwhile, the nature of threats and the methods used to address them differ fundamentally from the anomaly and data integrity issues. Hence, this work distinguishes itself by focusing on high-level data integrity, fake data injection, and anomalous behaviors in IoT systems. By employing a hybrid model that integrates optimized feature selection (ET-PBO) and a robust ensemble classification framework (SE-RAQD), this study offers a technically efficient and scalable solution for data anomaly detection in smart city and cyber–physical infrastructures.

1.1. Motivation

Traditional machine learning frameworks struggle to maintain accuracy and robustness in the presence of noise and class imbalance data in real-world applications. Furthermore, traditional models tend to overestimate and include irrelevant or redundant features. It is essential to obtain an intelligent algorithm that diminishes dimensionality by ensuring data integrity. These limitations are addressed by a hybrid model that merges the Elite Tuning-based Polar Bear Optimization (ET-PBO) algorithm, which performs superior feature selection, with a Stacking Ensemble-based Random AdaBoost Quadratic Discriminant (SE-RAQD) model for robust classification. The ET-PBO model mimics polar bear behavior, balancing the exploration and exploitation phase results in selecting relevant features. Furthermore, SE-RAQD incorporates the complementary strengths of AdaBoost, QDA, and Random Forest with a stacking ensemble framework that enhances detection performance by diminishing noise and imbalanced and high-dimensional systems. This combined model can achieve accurate, understandable, and resilient fake data detection that secures data environments.

1.2. Novelty

The Introduction of Elite-Tuning-based Polar Bear Optimization (ET-PBO) for Feature Selection: The proposed method integrates a novel Polar Bear Optimization algorithm called Elite-Tuning-based PBO (ET-PBO) that prioritizes top-performing solutions in the process of improving convergence speed and avoids falling into the local optimal solution. Thus, the overall performance is improved, with highly effective feature subset selection, by using a hybrid strategy that minimizes the dimensionality of high-dimensional data.

Stacking-Based Ensemble Framework for Improved Generalization: The classification stage incorporates an effective ensemble model named Stacking-Ensemble-Based Random AdaBoost Quadratic Discriminant (SE-RAQD), which integrates three base learners: AdaBoost, Quadratic Discriminant Analysis (QDA), and Random Forest (RF). These models capture different decision boundaries: AdaBoost learns from difficult samples, QDA models class distributions probabilistically, and RF ensures robustness through bagging. The predictions of all three models are integrated through the stacking method, where a meta-learner learns to optimally weight their outputs. This model significantly enhances effectiveness and generalization capability, especially in complex, noisy, and imbalanced environments.

End-to-end Integration to Detect Fake Data: The proposed model presents a fully integrated end-to-end pipeline that starts with data collection and ends with fake data classification. Feature selection and classification are not treated as disjoint tasks; instead, the ET-PBO module optimizes feature subsets specifically tailored to enhance SE-RAQD's classification accuracy.

Applicability to Real-World Noisy and Imbalanced Scenarios: The model is explicitly designed to handle imbalanced and noisy data, a common challenge in cybersecurity, IoT, and smart city infrastructures. The integration of ensemble diversity and adaptive optimization renders the model highly applicable to real-time fake data detection systems that require rapidly produced, accurate, and interpretable results.

1.3. Contributions

The significant contributions of this work are as follows.

Novel Feature Selection Algorithm: Elite-Tuning-Based Polar Bear Optimization (ET-PBO). In this study, ET-PBO was developed to effectively balance exploration and exploitation while searching for optimal feature subsets, reducing dimensionality while preserving crucial data. Overall, the classification performance is enhanced, and computational complexity is reduced.

Introducing Robust Stacking Ensemble Classifier: SE-RAQD. In this paper, a Stacking Ensemble-Based Random AdaBoost Quadratic Discriminant (SE-RAQD) model is introduced, leveraging the merits of three different base learners: Adaptive Boosting (AdaBoost), Quadratic Discriminant Analysis (QDA), and Random Forest (RF). Overall, the SE-RAQD model enhances resilience against noise, overfitting, and class imbalance.

Integration of Feature Selection and Ensemble Classification for Fake Data Detection: The proposed framework integrates ET-PBO for feature selection with the SE-RAQD ensemble classifier to develop a highly effective and scalable fake data detection system. Before classification, this integration optimizes feature relevance, thereby improving the predictive accuracy and generalization in real-world noisy and high-dimensional data scenarios.

Comprehensive Evaluation of Benchmark Datasets: During fake data detection, multiple benchmark datasets were evaluated using the proposed model. The simulation findings demonstrate significant improvements over state-of-the-art methods, highlighting the model's practical applicability.

The remaining sections are organized as follows. Section 2 reviews past studies related to fake data detection in smart cities. Sections 3 and 4 present the proposed methodology. Sections 5 and 6 analyze the experimental results, and Section 6 concludes the paper, outlining the scope of future studies.

2. Literature Review

Asavisanu et al. (2025) [12] developed the Cooperative Autonomy Trust and Security (CATS) model to enhance security in Vehicle-to-Everything (V2X) communications. The CATS model's performance was validated using real-time traffic data, and the results highlighted superior performance in detecting misbehaving vehicles. Sani et al. (2024) [13] presented a blockchain-enabled intrusion detection system (BIDS) to improve security and reliability within the Internet of Vehicles (IoV). The mobility patterns were generated in BIDS and used to enhance the model, with effective performance in terms of detecting malicious vehicles accurately.

He et al. (2023) [14] generated an unsupervised reward learning approach using Long Short-Term Memory (LSTM) with a Q-learning algorithm (LSTM-Q) to enhance robustness in anomaly detection with minimum traffic issues. In real-time data validation, the city of Brisbane was used to analyze the efficiency of the LSTM-Q model. Nayak et al. (2023) [15] introduced a Machine Learning-Based Misbehavior Detection System (ML-MDS) for Cognitive-Software-Defined Multimedia Vehicular (CSDMV) networks in smart cities. The UNSW_NB-15 dataset was used for analysis, and it was found that the ML-MDS model achieved superior detection performance in the detection of malicious vehicles.

Ji et al. (2024) [16] presented a hybrid approach that merges the Cyber-Physical Digital Twin with a deep learning-based Intrusion Detection System (IDS). The response strategies were developed by the Digital Twin, and the response was effectively carried out with a real-time monitoring process. Saleem et al. (2022) [17] discussed enhancing traffic flow in smart cities using a Fusion-Based Intelligent Traffic Congestion Control System for VNs (FITCCS-VN). The advanced model offered better decisions with respect to avoiding traffic jams. The experimental results revealed that the developed FITCCS-VN model achieved superior performance in terms of enhanced traffic flow.

AlJamal et al. (2024) [18] addressed fake clients by developing a simulated smart city model using the program Netsim. Wireshark was used to capture data flow and store files in CSV format. The findings from experiments demonstrated that the smart city model attained better performance in terms of detecting fake customers. Ajao et al. (2023) [19] utilized Petri Net and Genetic-Algorithm-Based Reinforcement Learning (GARL) to enhance a robust security framework for smart cities. The simulation results revealed that this model achieved superior performance in terms of security. Ragab et al. (2025) [20] implemented Advanced Artificial Intelligence with a Federated Learning Framework for Privacy-Preserving Cyberthreat Detection (AAIFLF-PPCD), which enhanced the privacy of IoT users in smart cities.

Mishra et al. (2024) [21] developed the Hybrid Deep Learning Long Short-Term Memory–Support Vector Machine (LSTM-SVM) algorithm to safeguard various transactions in smart cities. The reptile search algorithm was used to select the crucial features, which were then allowed into the blockchain-based distributed network using the industrial gateway. The results highlighted that the developed LSTM-SVM attained more effective performance in securing transactions.

To authenticate transactions carried out by businesses, Kumar and Kumar (2024) [22] developed a modified deep neural network (M-DN) classifier. The data was stored in a private blockchain using the Elliptic Curve Integrated Encryption Scheme (ECIES) after being authenticated as normal. Otherwise, the data was passed to the hybrid LSTM-XGBoost

classifier for the detection of attack type. The evaluations showed that the developed model exhibited better performance in terms of accuracy and encryption time.

Arif et al. (2023) [23] developed a model based on a Generative Adversarial Network (GAN) and Deep Reinforcement Learning (DRL) to generate fully functional adversarial malware samples that can evade detection by machine learning-based malware detectors while preserving their malicious behavior. The unfiltered feature set of the malware dataset was fed into the GAN. The findings revealed that the developed GAN-DRL model attained more effective performance.

Research Gaps and Limitations

Although various models have been developed to improve security, traffic management, and anomaly detection in smart city and vehicular networks, several limitations persist. Many of the existing models lack scalability and generalization in heterogeneous environments with diverse data sources. Real-time responsiveness is another important concern, as some methods are computationally intensive, making them less suitable for dynamic environments. Furthermore, the limited integration of feature optimization and adaptive learning strategies reduces detection efficiency, rendering existing algorithms ineffective against emerging threats and fake data patterns. Privacy-preserving approaches face difficulty in balancing detection accuracy with user data confidentiality. These gaps highlight the need for a robust and intelligent framework that can achieve accurate, real-time fake data detection while being efficient, adaptive, and privacy-aware across various smart city infrastructures.

3. Intelligent Urban Infrastructure Framework—A System Design

The smart city architecture includes three interconnected layers, as depicted in Figure 1.

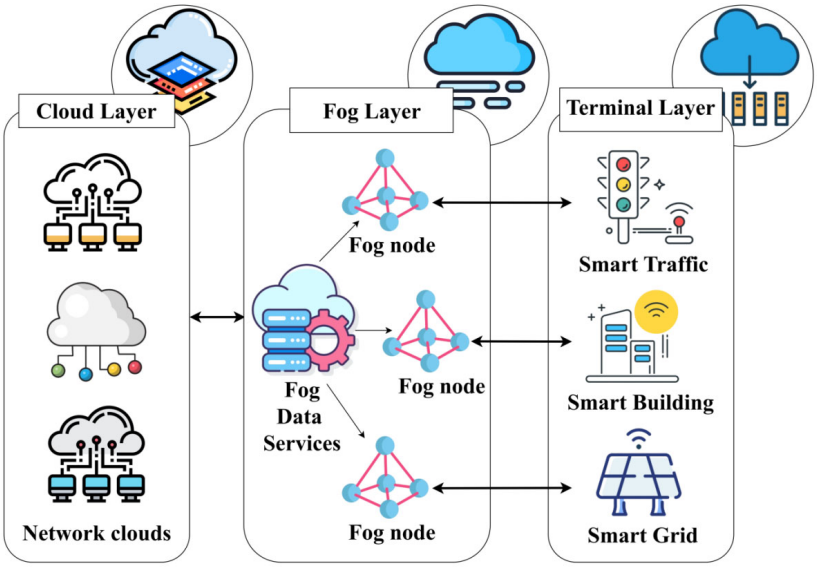


Figure 1. A smart city model illustrating the three interconnected layers, showing data flow from network clouds through fog nodes to smart traffic, smart building, and smart grid applications.

Cloud Layer: Centralized Intelligence and Storage Backbone

The cloud layer functions as the centralized core of the smart city infrastructure. It is responsible for the long-term storage, advanced analytics, decision support, and global coordination of data collected from across the urban environment. The cloud hosts high-performance computing resources, databases, machine learning models, and visualization tools that facilitate macro-level planning and policy execution.

- In the cloud layer of the proposed smart city architecture (Figure 1), integrating a Verifiable Query Layer (VQL) significantly enhances the efficiency, transparency, and security of data access and management.
- VQL enables verifiable cloud query services, allowing users or smart applications to query large volumes of sensor data stored in the cloud while receiving cryptographic proof of the correctness and completeness of query results.
- This capacity is especially critical in blockchain-integrated systems, where VQL can act as a trusted interface between off-chain storage and on-chain smart contracts or decentralized applications.
- By deploying VQL within the cloud layer, smart city stakeholders can make auditable, efficient, and privacy-preserving queries.
- This framework also aligns with federated environments, where multiple data contributors require proof of query fairness and consistency.

Fog Layer: Localized Processing and Real-Time Response

The fog layer, also referred to as the edge-computing interface, acts as an intermediary between the terminal layer and the cloud. It consists of distributed micro-data centers or edge nodes strategically positioned across different zones of the city. These nodes handle time-sensitive computations, local decision making, and preliminary analytics before passing refined data to the cloud.

Terminal Layer: Perception and Data Acquisition

The terminal layer is composed of an extensive network of IoT devices deployed throughout urban infrastructure. These devices include sensors, actuators, RFID tags, smart meters, cameras, and wearable devices embedded in systems such as public transportation, utilities, street lighting, buildings, and personal environments.

4. Proposed Methodology

Figure 2 illustrates the conceptual framework of the proposed methodology designed to detect fake data accurately, particularly in noisy datasets common to smart city environments, IoT ecosystems, and cyber-physical systems. Five key stages, namely, Data Collection, Preprocessing, Feature Selection, Classification, and Prediction of Fake Data, are incorporated into this hybrid framework by applying a novel combination of Elite-Tuned Polar Bear Optimization (ET-PBO) for feature selection and the Stacking-Ensemble-based Random AdaBoost Quadratic Discriminant (SE-RAQD) model for classification.

4.1. Data Collection

In this study, the following four datasets were employed in detecting fake data.

Dataset Description

UNSW-NB15 dataset: This dataset (https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15 (accessed on 7 August 2025)) was published by the Lab of the Australian Centre for Cyber Security, comprising 44 features that are numerical or nominal values. The dataset includes 257,673 records and 10 types of attacks.

CICIDS2017 Dataset: This intrusion detection evaluation dataset (https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset (accessed on 7 August 2025)) comprises nearly 2,830,540 distinct instances along with 15 classes and 83 features.

CIC IoT Dataset 2023: This dataset, which focuses on IoT network traffic, was primarily developed for intrusion detection (https://www.kaggle.com/datasets/akashdogra/cic-iot-2023 (accessed on 7 August 2025)). It has seven different classes: Benign, DDoS, DoS, Mirai, MITM (Man-in-the-Middle), Reconnaissance, and Theft. The dataset, which is used

to develop and evaluate security solutions for IoT environments, contains both normal and malicious traffic instances.

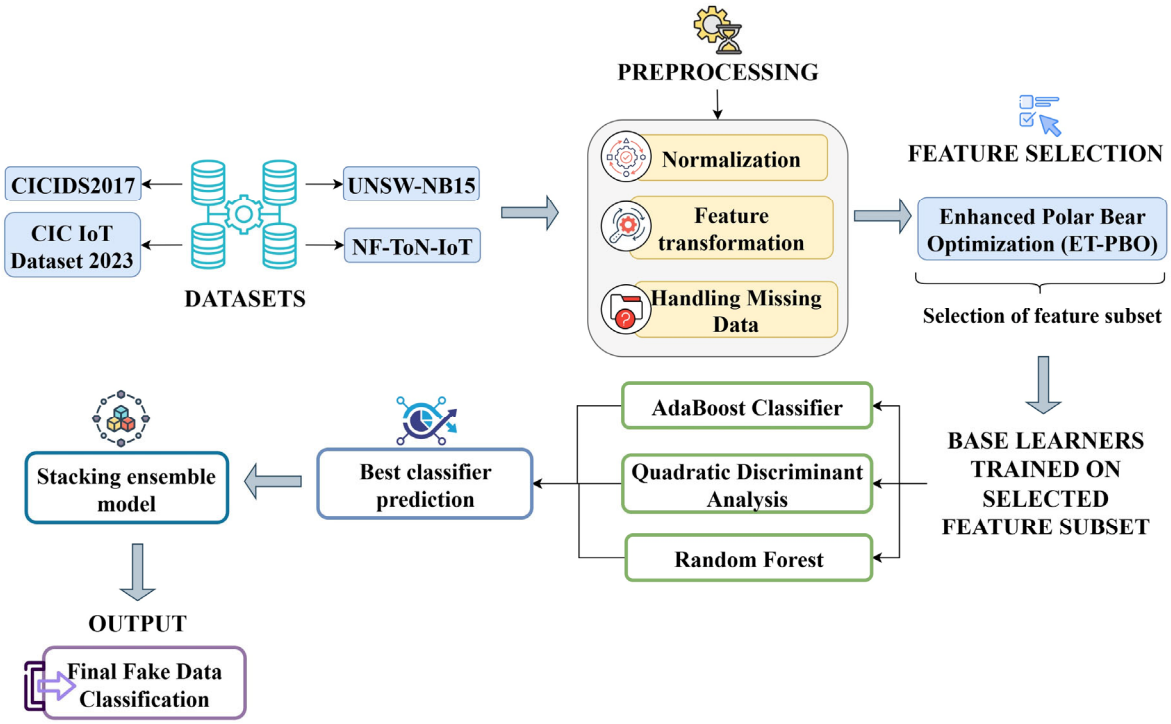


Figure 2. Conceptual framework of the proposed methodology.

NF-ToN-IoT dataset: This dataset (https://www.kaggle.com/datasets/dhoogla/nftoniot (accessed on 7 August 2025)) captures real-world traffic from various IoT devices, and it contains both normal traffic and various attacks, including DDoS, DoS, espionage, injection, password cracking, backdoor, XSS, MITM, ransomware, and normal. This dataset is helpful in research on smart cities and industrial IoT infrastructures, assisting in developing advanced security mechanisms through anomaly detection techniques.

4.2. Data Preprocessing

Data preprocessing ensures that raw data collected from various sources is converted into a format suitable for analysis and model training. This step typically involves several key approaches, which are described below.

Normalization: For processing, the data transformation of the min–max scaling method [24] is used. It is expressed in the equation below.

$$I_y = \frac{D_y - \min\ (D_y)}{\max\ (D_y) - \min\ (D_y)} \tag{1}$$

where $\min\ (D_y)$, $\max\ (D_y)$, I_y, and D_y represent the minimum, maximum, normalized, and input values, respectively.

Handling Missing Data: In this process, the collected data is effectively handled, and information leakage is reduced [25]. Furthermore, sufficient data—obtained using the averaging process—is found, and missing data is identified. This process yields an average value that can be used to calculate the average range of the data obtained from the specified factors. The mathematical expression is given as follows.

$$B_{FILLED} = \frac{1}{E}\sum_{j=1}^{E} B_j \tag{2}$$

where the anticipated ranges obtained for gap filling are denoted by B_{FILLED}, and the observed number of non-missing values is represented by B_j and E, respectively.

Feature transformation: This includes approaches such as one-hot encoding [26] for categorical variables, converting categories into binary matrices, and other transformations that improve the representational power of the data. It also includes dimensionality reduction approaches to simplify the dataset while retaining essential information.

For a categorical feature D with m categories, m binary features are created.

$$D' = \begin{cases} 1 & if\ category\ is\ present \\ 0 & otherwise \end{cases} \tag{3}$$

4.3. Feature Selection Using Elite-Tuning-Enhanced Polar Bear Optimization (ET-PBO)

In smart city network analysis, high-dimensional datasets face a few major difficulties, including the presence of irrelevant or redundant features. This degrades the classification performance of a model because of overfitting and increased computational complexity. These difficulties are addressed by the proposed approach via integrating an Elite-Tuning-enhanced Polar Bear Optimization (ET-PBO) algorithm. Let us consider each solution (or candidate) as a "polar bear" searching for food in a large, icy region (i.e., the feature space). The current location of each bear represents a selected subset of features. The bear moves by adjusting its position based on two strategies: (1) exploration, which allows the investigation of new areas, and (2) exploitation, which focuses more on the areas already known to be good.

Let us consider that the dataset has a total of 10 features. One candidate solution might be a binary vector like 1010101010, where 1 means a feature is selected, and 0 means the features are not selected. ET-PBO evaluates this solution based on how well a classifier performs using these features. If better-performing candidates (bears) select different subsets like 1111000000, the current bear may move closer to that configuration using controlled randomness. Elite tuning ensures that only the top-performing bears guide the search, accelerating convergence and avoiding poor local optima. This process repeats iteratively until the best subset of features is found, minimizing redundancy while maximizing classification accuracy. The final selected features are then passed to the SE-RAQD classifier stage.

This algorithm can select the most significant and relevant feature subset by balancing the exploration and exploitation phases. Overall, the classification accuracy is enhanced while reducing dimensionality.

Consider a dataset with e features denoted by the set $\Im = \{g_1,\ g_2, \ldots\ldots,\ g_e\}$ [27]. In the search space, each candidate solution is encoded as a binary vector.

$$p = [p_1,\ p_2,\ \ldots,\ p_e], \qquad p_j \in \{0,\ 1\} \tag{4}$$

where $p_j = 1$ represents the feature inclusion g_j, and $p_j = 0$ represents its exclusion. The fitness function $F(p)$ computes each candidate solution p, and it is expressed using the equation below.

$$F(p) = \beta.Accuracy(p) - \alpha.\frac{\sum_{j=1}^{e} p_j}{e} \tag{5}$$

where the classification accuracy attained by classifier training is represented by $Accuracy(p)$. The number of selected features is denoted by $\sum_{j=1}^{e} p_j$. The weighting factors are denoted by α and β.

The ET-PBO algorithm randomly initializes several candidate feature subsets $\{p_1,\ p_2, \ldots\ldots,\ p_N\}$ to simulate the distribution. Each candidate repeatedly updates its position by simulating polar bears' hunting strategies, including path tracking and stealth

movement, enabling efficient navigation through the feature subset space, and this process is improved by elite tuning, in which the convergence speed is increased. Elite tuning also assists in avoiding problems relating to local optima. The equation below expresses the position update of each candidate p_i in the population.

$$p_i^{u+1} = p_i^u + \lambda.\Delta(p_i^u,\ p_{elite}) + \gamma.\aleph \tag{6}$$

where the position of candidate i in the $u\ th$ iteration is denoted by p_i^u. The position vector from the elite set is denoted by p_{elite}. The control parameters are represented by λ and γ. The random vector that introduces stochasticity is denoted by $\aleph$. The difference operator modeling the movement towards elite solutions is denoted by $\Delta(.)$.

Since this problem is binary, after updating, the continuous position vector is mapped back to a binary solution using a sigmoid transfer function $\sigma(.)$, and this process is followed by thresholding.

$$Q\left(p_j^{u+1} = 1\right) = \sigma\left(y_j^{u+1}\right) = \frac{1}{1 + f^{-y_j^{u+1}}} \tag{7}$$

where the continuous position for the feature j before binarization is y_j^{u+1}. The probability of the binary solution p_j^{u+1} for the $i\ th$ feature at iteration $u+1$ is denoted by $Q\left(p_j^{u+1} = 1\right)$. This algorithm iterates until a maximum number of iterations U is reached.

The ET-PBO approach minimizes dimensionality by removing unnecessary features, thereby improving generalization capacity. The binary vector is the final output of ET-PBO, denoting the optimal feature subset. This vector, when used as input to the downstream Stacking Ensemble-Based Random Adaboost Quadratic Discriminant (SE-RAQD) model, maximizes classification accuracy while also reducing redundancy, noise, and irrelevant features. Furthermore, learning efficiency is enhanced by the ET-PBO approach, and it assists in detecting fake data or cyber anomalies in smart city environments. The algorithm for ET-PBO Algorithm 1 is given below. Figure 3 depicts a flowchart of the ET-PBO algorithm.

Algorithm 1: ET-PBO Algorithm for optimal feature selection

Input: Dataset with feature set, population size N, max iterations U
Initialize the population $Q = \{p_1,\ p_2, \ldots,\ p_N\}$ with random binary vectors
Evaluate fitness $g(p_j)$ for each p_i in Q
Initialize elite set with top solutions from Q
For u in range $(1,\ U+1)$:
    For each candidate p_i in Q:
        Select elite solution p_{elite} from F
        # Update position towards the elite and add random exploration
p_i continuous = $p_i + \lambda * (p_{elite} - p_i) + \gamma^* random\_vector()$
        # Apply the sigmoid transfer function and binarize
        for j in range (e):
            $prob = 1/(1 + \exp(-p_i\ continuous[j]))$
$p_i[j] = 1$ if $prob > random\_uniform(0,\ 1)$ $else$ 0
        Evaluate fitness $g(p_j)$
        Update the elite set F with top solutions from the current population
        Update P_{best} if better solution found
Return P_{best}
Output: Optimal feature subset P_{best}

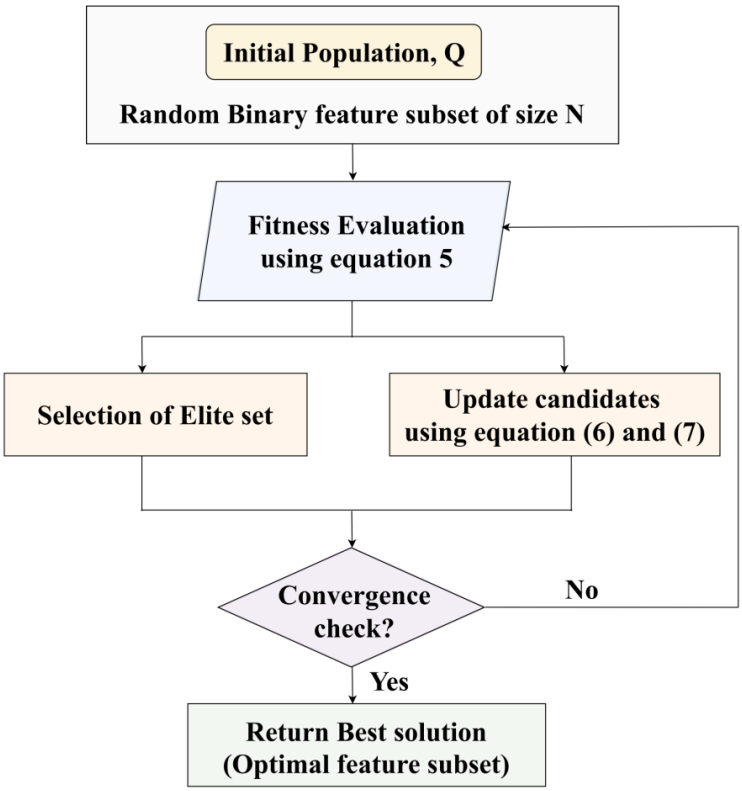


Figure 3. Flowchart of the ET-PBO algorithm.

4.4. A Novel SE-RAQD for Fake Data Detection in Smart Cities

The rapid expansion of smart cities facilitates intelligent automation across areas like traffic control, safety management, environmental tracking, and utility operations. Despite these advancements, the limited computational resources and diverse communication standards of IoT devices expose them to various cyber threats, including spoofing and denial-of-service (DoS) attacks, malicious data injection, and botnet-based intrusions. In this study, these complexities are addressed by introducing the Stacking Ensemble-Based Random AdaBoost Quadratic Discriminant (SE-RAQD) model that integrates the three base classifiers such as the ADA, QDA, and RF using the SEL strategy. This hybrid model enhances generalization and detection accuracy and effectiveness against adversary behaviors in IoT-based smart city networks.

4.4.1. AdaBoost Classifier

AdaBoost [28] is a sequential ensemble method that constructs a strong classifier by integrating a series of weak learners, typically decision stumps. In every iteration, AdaBoost assigns greater importance to the incorrectly classified instances by boosting their weights, compelling the next weak learner to concentrate more on these challenging examples. The equation below expresses the final decision function of AdaBoost.

$$G(y) = sign\left(\sum_{n=1}^{N} \beta_n g_n(y)\right) \tag{8}$$

The prediction of the $n\ th$ weak learner is $g_n(y)$, and its corresponding weight is represented by β_n. This weight is evaluated based on the classification accuracy.

$$\beta_n = \frac{1}{2}\ln\left(\frac{1-\Re_m}{\Re_n}\right) \tag{9}$$

where the classification error rate of g_n is denoted by $\Re_m$. The final binary class label is denoted by $G(y) \in \{-1, +1\}$. This mechanism ensures that the best-performing learners have a greater influence on the final outcome, making AdaBoost very effective in dealing with high-stakes situations.

4.4.2. Quadratic Discriminant Analysis (QDA)

QDA [29] allows for a quadratic decision boundary because the covariance is handled separately for each class. The probability of observing a sample $y \in \mathbb{R}^e$ given class $x = l$ is

$$Q(y|x = l) = \frac{1}{(2\phi)^{e/2}|\sum_l|^{1/2}} \exp\left(-\frac{1}{2}(y - \mu_l)^U \sum_l^{-1}(y - \mu_l)\right) \tag{10}$$

In the above equation, the mean vector and covariance matrix of class l are denoted by $\mu_l \in \mathbb{R}^e$ and $\sum_l \in \mathbb{R}^{e \times e}$, respectively. The total number of features is denoted by $Q(y|x = l)$. The Bayes theorem is used to evaluate the posterior probability $Q(y|x = l)$. The class label $\hat{y} = \text{argmax}_l Q(x = l \backslash y)$ is assigned to attain the final classification. The strength of QDA relies entirely on its ability to model class-specific distributional properties, making it very useful in a domain where class variances are heterogeneous.

4.4.3. Random Forest (RF)

RF [30] is a bagging-based ensemble method where two levels of randomness are employed. The equation below expresses the prediction function for RF.

$$\hat{y} = \text{mode}\{g_u(y)\}_{u=1}^U \tag{11}$$

where the output of the u^{th} decision tree in an ensemble of U trees is denoted by $g_u(y)$. This method minimizes the overfitting issues by averaging the multiple low biases in which the stable and accurate classifier is developed.

4.4.4. Stacking Ensemble Mechanism

The SE-RAQD framework uses a stacking ensemble strategy to aggregate the predictions from the three base learners into a high-level result [31]. After training AdaBoost, QDA, and Random Forest independently on the same input feature vector, their respective outputs form a meta-feature vector:

$$C = \left[g_{ADA}(y),\ g_{QDA}(y),\ g_{RF}(y)\right] \in \mathbb{R}^3 \tag{12}$$

The immediate decisions are encoded by this meta-feature vector and act as the meta-learner input $\phi(.)$, and it is trained to integrate all the outputs. The equation below expresses the final classification.

$$\hat{x} = \phi(C) \tag{13}$$

4.4.5. Ensemble Synergy and Robustness

The design of SE-RAQD takes advantage of the complementary strengths of its block learners. AdaBoost promotes adaptive learning by focusing on complex models; QDA models the probabilistic structure of each class with a designed covariance matrix; and RF ensures stability and generalization through bagging. When these models are stacked, their decision patterns are combined by the meta-learner, resulting in a unified classifier capable of handling class imbalance, noise, and complex decision boundaries. Logistic regression was selected as the meta-classifier for the SE-RAQD model due to its balance of simplicity, interpretability, and efficiency. It performs well when combining probabilistic outputs from diverse base learners such as AdaBoost, QDA, and Random Forest. Logistic regression

yielded comparable accuracy with faster convergence and lower computational overhead, making it suitable for real-time smart city environments. Hence, it was chosen to preserve both high performance and low latency.

Algorithm 2 for fake data detection using ET-PBO + SE-RAQD is given below. Figure 4 illustrates the flowchart for the SE-RAQD model.

Algorithm 2: Fake Data Detection using ET-PBO + SE-RAQD

Input: Raw dataset with features G and labels X

Step 1: Data Preprocessing
Normalize features using Min-Max scaling (Equation (1))
Handling missing data (Equation (2))
Apply One-Hot Encoding to categorical features (Equation (3))

Step 2: Feature Selection using ET-PBO
Initialize the population of feature subsets $P = \{p_i\}$
for u in range $(\max\_iterations)$:
    Evaluate fitness $G(p_i)$ for each subset (Equation (4))
    Select elite subsets based on fitness
    Update population by polar bear-inspired moves (exploration + exploitation)
Select the best feature subset (P_{best})

Step 3: Model Training with Selected Features
Train base classifiers:
    ADA = AdaBoost trained on features in P_{best}
    QDA = Quadratic Discriminant Analysis on P_{best} S_best
    RF = Random Forest on P_{best} S_best

Step 4: Stacking Ensemble
For each instance x in the validation set:
    Get predictions $[g_{ADA}(y),\ g_{QDA}(y),\ g_{RF}(y)]$
Form meta-features $C = [g_{ADA}(y),\ g_{QDA}(y),\ g_{RF}(y)]$
Train meta-classifier ϕ on C and true labels

Step 5: Prediction on Test Set
For each test instance y_{test}:
    Obtain base predictions $g_{ADA}(y_{test}),\ g_{QDA}(y_{test}),\ g_{RF}(y_{test})$
    Form $C_{test} = [g_{ADA}(y_{test}),\ g_{QDA}(y_{test}),\ g_{RF}(y_{test})]$
    Predict label $X_{hat} = \phi(C_{test})$

Return X_{hat}
Output: Predicted labels X_{hat} for test data

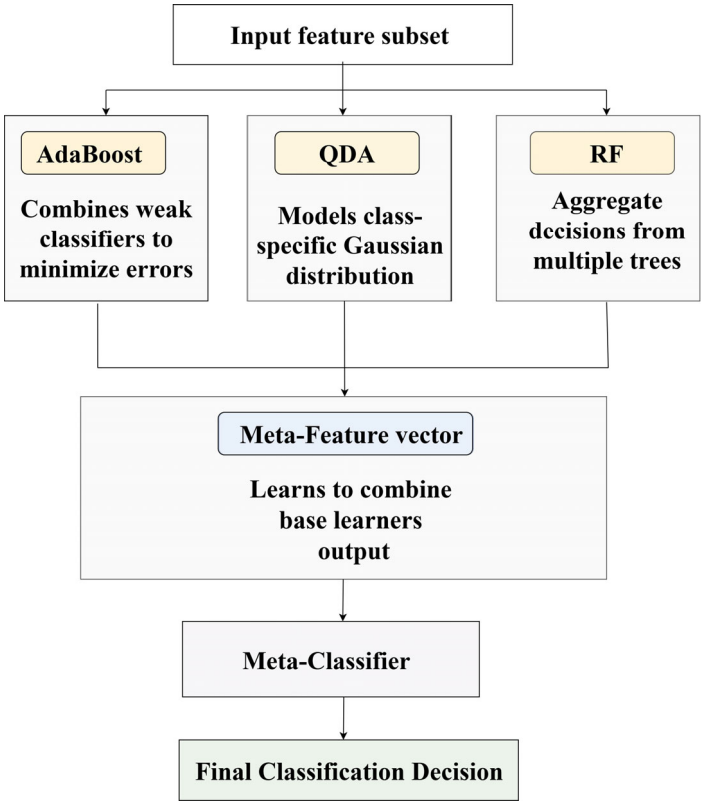


Figure 4. Flowchart for SE-RAQD model.

5. Experimental Results

In this section, several evaluation metrics are employed to assess the performance of the proposed model.

5.1. Experimental Setup

An Intel® Core™ i7-5960X CPU (Intel Corporation, Santa Clara, california (CA), United States of America (USA)), 32 GB of RAM, and an NVIDIA GeForce RTX 2060 GPU (NVIDIA Corporation, Santa Clara, CA, USA) were installed on a high-performance workstation for the evaluation and implementation of the results. Python 3.8.10 was used to conduct all experiments. To ensure fair and consistent evaluation across all datasets, each dataset was divided into 80% training and 20% testing subsets. Splitting was conducted using stratified sampling to preserve the original class distributions. This data-splitting process ensures that both majority and minority classes are represented in both the training and testing sets. In addition, all preprocessing steps that include normalization, feature encoding, and missing value handling were performed upon training the data. The learned transformation parameters were applied to the test set to prevent any form of data leakage.

5.2. Parameter Settings

Table 1 lists the parameter settings for the proposed model that includes ET-PBO and SE-RAQD components.

Table 1. Parameter configuration with optimal values.

Parameters of ET-PBO Model and SE-RAQD Model	Values
Population size	50
Maximum number of iterations	100
Control parameters	0.6, 0.3
Sigmoid threshold	0.5
Total number of elite candidates	5
Number of estimators	100
Learning rate	0.1
Meta Learner	Logistic Regression

5.3. Evaluation Metrics

The following key evaluation metrics were used to evaluate the performance of the proposed model.

Accuracy: This is the proportion of correctly detected real and fake data instances out of the total data processed in a smart city system.

$$Accuracy = \frac{T_{po} + F_{po}}{T_{po} + T_{ne} + F_{po} + F_{ne}} \tag{14}$$

Precision: This reflects the system's reliability in detecting fake data accurately, highlighting how many of the instances labeled as fake are truly fake.

$$\text{Pr}ecision = \frac{T_{po}}{T_{po} + F_{po}} \tag{15}$$

Recall: This identifies the fraudulent data accurately by computing the ratio of true fake instances that are identified successfully.

$$\text{Re}call = \frac{T_{po}}{T_{po} + F_{ne}} \tag{16}$$

Specificity: This accurately recognizes the genuine data and minimizes incorrect labels.

$$Specificity = \frac{T_{ne}}{T_{ne} + F_{po}} \tag{17}$$

F1-score: This identifies fraudulent data during imbalanced dataset handling.

$$F1 - score = 2 \frac{\text{Pr}ecision * \text{Re}call}{\text{Pr}ecision + \text{Re}call} \tag{18}$$

Kappa score: This provides a reliable measure of classifier performances in the case of smart city applications.

$$\kappa = \frac{A_O - A_E}{1 - A_E} \tag{19}$$

Throughput: This is defined as the proportion of total processed data by the total time taken.

$$Throughput = \frac{Total\ data\ processed}{Total\ time\ taken} \tag{20}$$

Latency: Latency refers to the time delay between when data is received and when the system generates an output or decision. In real-time applications, lower latency indicates

a faster response to incoming data, which is crucial for promptly identifying fake or malicious data.

$$Latency = \frac{Total\ time\ \ taken}{Total\ data\ processed} \tag{21}$$

CPU utilization: CPU utilization denotes the amount of processing power used during fake data detection and contributes to assessing how resource-intensive the model is on smart city infrastructures.

$$CPU\ utilization = \frac{CPU\ time\ used}{Total\ CPU\ time} \times 100 \tag{22}$$

5.4. Performance Analysis

The key performance metrics include the confusion matrix, which summarizes true and false predictions, along with the AUC-ROC and Precision–Recall curves, which reflect the model's classification quality. Table 2 depicts the performance evaluation on four diverse datasets with respect to diverse measures.

Table 2. Performance on four benchmark datasets.

Metrics	Datasets			
	UNSW-NB15	**CICIDS2017**	**CIC IoT Dataset 2023**	**NF-ToN-IoT**
Accuracy	98.78%	98.25%	97.98%	97.55%
Precision	98.75%	98.16%	97.90%	97.50%
Recall	98.66%	98.33%	97.55%	97.42%
Specificity	98.72%	98.10%	97.80%	97.60%
F1-score	98.70%	98.24%	97.72%	97.45%
Kappa score	98.76%	98.55%	97.80%	97.30%
Throughput	980 (instances/s)	950 (instances/s)	890 (instances/s)	920 (instances/s)
Latency	16 ms	18 ms	22 ms	23 ms
CPU utilization	18%	22%	25%	27%

Figure 5 provides the results of the accuracy analysis of the proposed model based on all four datasets. Figure 5a shows that the training accuracy of the model is more effective when it is applied to on all four datasets. Specifically, the UNSW-NB15 dataset led to slightly higher training accuracy than the others. Similarly, Figure 5b illustrates the model's testing accuracy on all datasets. This graph demonstrates that the testing accuracy of the model is higher when applied to the UNSW-NB15 dataset.

Figure 6 depicts the loss analysis of the proposed model with respect to all four datasets. The training loss graph is depicted in Figure 6a, where the training loss of the model is low on all four datasets. Specifically, the UNSW-NB15 dataset led to lower training loss than all the others. Figure 6b demonstrates that the model achieved low testing loss based on all four datasets, where the UNSW-NB15 dataset led to the lowest testing loss relative to the other datasets.

Figure 7a illustrates the results of the AUC-ROC analysis of the proposed model on all four benchmark datasets. From this graphical representation, it can be gleaned that the model attains higher AUC-ROC values when applied to all four datasets, whereas the UNSW-NB15 dataset led to higher AUC-ROC values relative to all the other datasets. Figure 7b depicts the precision–recall analysis of the proposed model on all the datasets. This graphical analysis highlights that the model achieved higher precision–recall values on all the datasets. Specifically, the UNSW-NB15 dataset led to more effective values than all the other datasets.

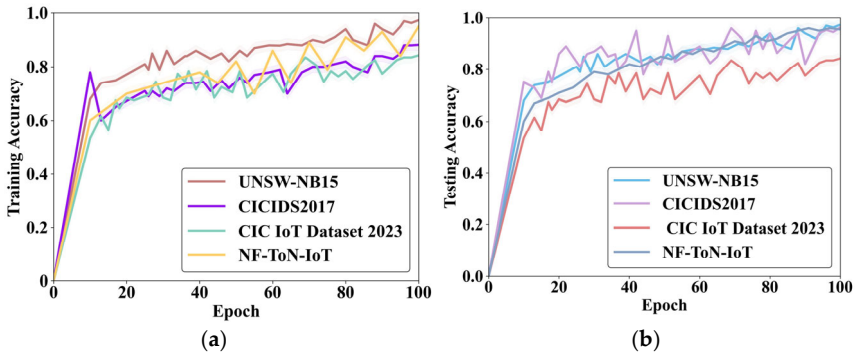


Figure 5. Accuracy analysis of the proposed SE-RAQD model on all four benchmark datasets. (**a**) Training accuracy shows consistently high performance across datasets, with the UNSW-NB15 dataset achieving the highest value, indicating strong learning capability. (**b**) Testing accuracy mirrors the training trends, demonstrating that the model generalizes well. UNSW-NB15 and CICIDS2017 datasets yield higher test accuracy compared to CIC IoT 2023 and NF-ToN-IoT, reflecting their more stable or well-structured feature spaces.

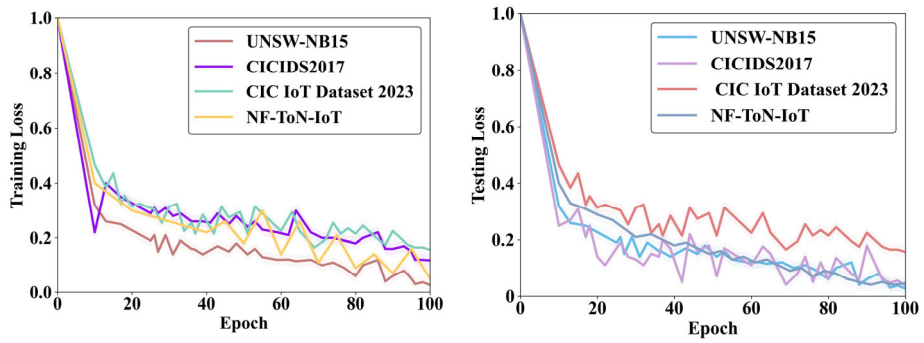


Figure 6. Training and testing loss curves for the SE-RAQD model. (**a**) Training loss steadily decreases across all datasets, with the UNSW-NB15 dataset achieving the lowest loss, suggesting efficient learning. (**b**) Testing loss remains low and stable, particularly for UNSW-NB15 and CICIDS2017, further confirming the model's robustness and minimal overfitting. The slightly higher test loss for the NF-ToN-IoT dataset may reflect greater variability and noise in its feature space.

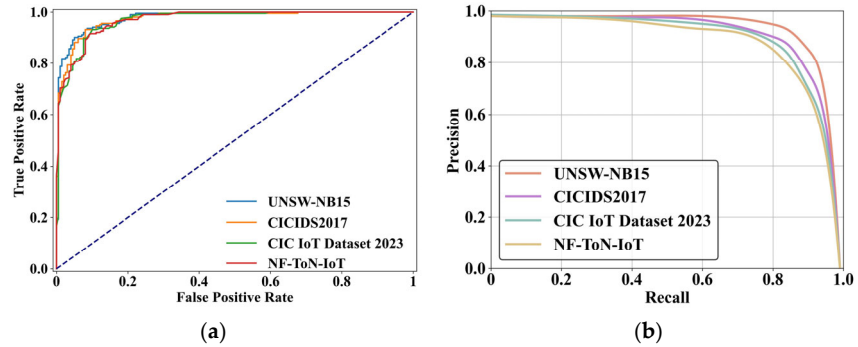


Figure 7. (**a**) Evaluation of the proposed model's classification quality using (**a**) AUC-ROC and (**b**) precision–recall curves. In both plots, the SE-RAQD model achieves near-perfect separation, especially on the UNSW-NB15 dataset. The CICIDS2017 and CIC IoT 2023 datasets also show strong curves, while NF-ToN-IoT shows slightly lower values, likely due to higher data imbalance and noise. These curves validate the model's effectiveness in correctly identifying both positive and negative classes across varied cyber-attack scenarios.

Figure 8 illustrates the fitness score analysis, depicting the fitness score of ET-PBO and PBO. This graphical representation reveals that the fitness score of ET-PBO is 0.93, whereas the fitness score achieved by PBO is 0.88. The fitness score of ET-PBO is higher than that of PBO, indicating optimal performance.

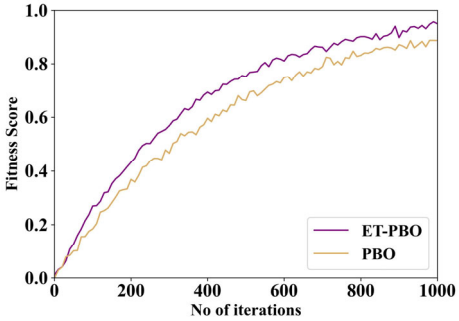
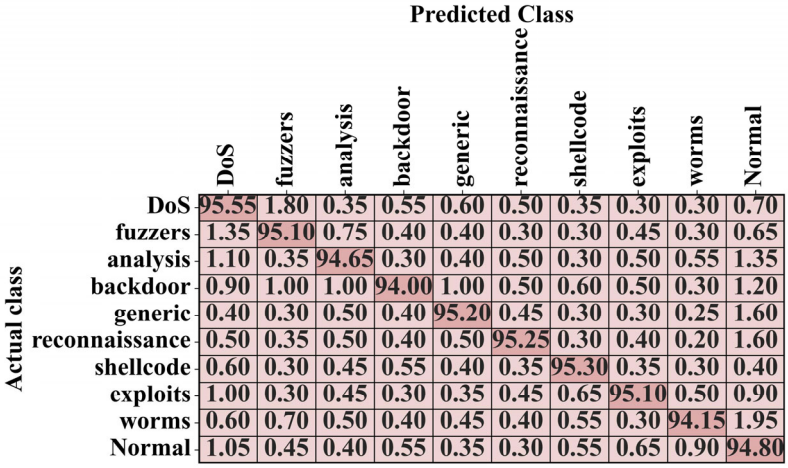
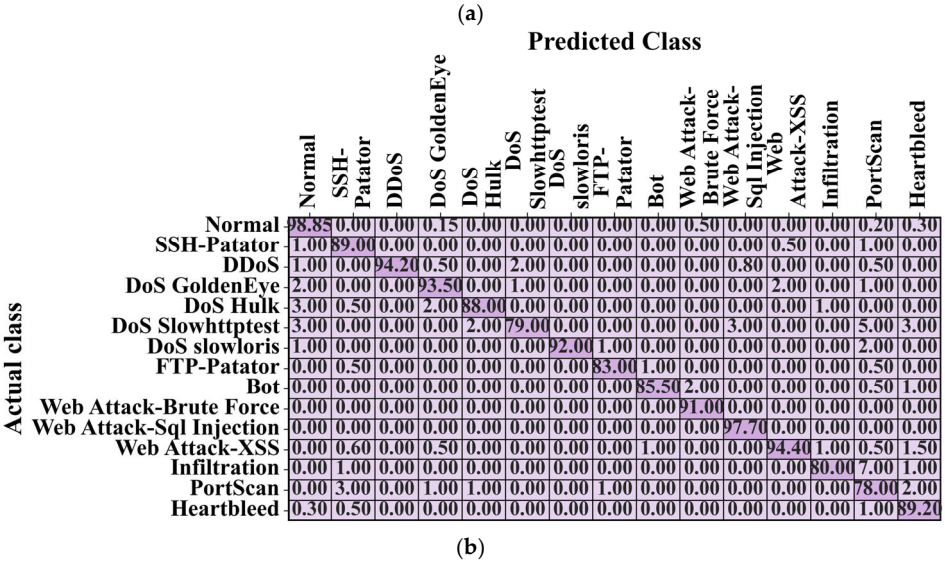


Figure 8. Fitness score analysis.

Figure 9a–d depicts the confusion matrix for four different datasets, including the ISIC Skin Disease Image Dataset Labeled, Skin Disease Dataset, Ham10000, Indian Skin Disease Dataset, Skin Cancer ISIC 2020, and Skin Disease Classification datasets. From these confusion matrices, the performances can be determined by comparing the actual class and predicted classes.



(**a**)



(**b**)

Figure 9. *Cont.*

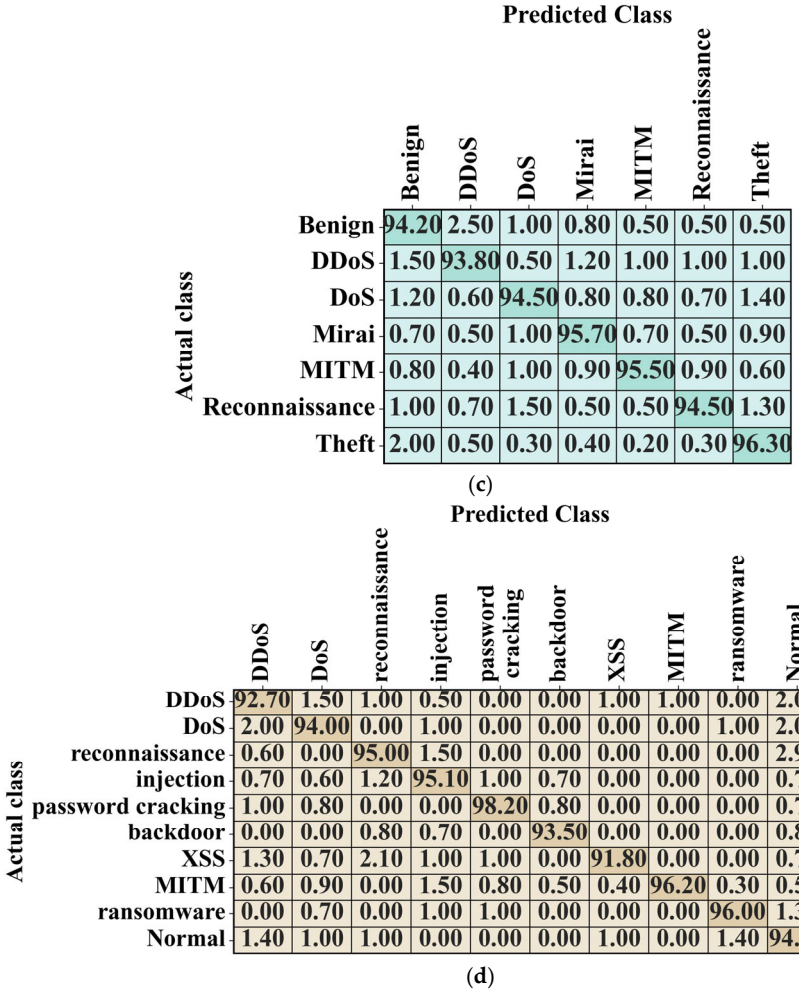


Figure 9. Confusion matrix for (**a**) UNSW-NB15, (**b**) CICIDS2017, (**c**) CIC IoT Dataset 2023, and (**d**) NF-ToN-IoT.

5.5. Comparative Analysis

Figure 10a–d illustrate the results of a comparative analysis of different existing models, including the LSTM-SVM, M-DNN, AAIFLF-PPCD, GARL, GAN-DRL, and proposed SE-RAQD models, on four benchmark datasets. Figure 10a highlights that the SE-RAQD model achieved higher performance than all the other models, with an accuracy of 98.78%, a precision of 98.75%, a recall of 98.66%, a specificity of 98.72%, an F1-score of 98.70%, and a kappa score of 98.76% on the UNSW-NB15 dataset. Based on the CICIDS2017 dataset, the SE-RAQD model achieved higher accuracy, precision, recall, specificity, and F1 and kappa scores, yielding values of 98.25, 98.16, 98.33, 98.10, 98.24, and 98.55%, respectively, as shown in Figure 10b. Figure 10c highlights the higher efficiency of the SE-RAQD model, with an accuracy of 97.98%, a precision of 97.90%, a recall of 97.55%, a specificity of 97.80%, an F1-score of 97.72%, and a kappa score of 97.80% on CIC IoT Dataset 2023. Figure 10d shows that the SE-RAQD model achieved higher performance than all the other models, with an accuracy of 97.55%, a precision of 97.50%, a recall of 97.42%, a specificity of 97.60%, an F1-score of 97.45%, and a kappa score of 97.30% on the NF-ToN-IoT dataset. These comparative results showed that the SE-RAQD model performs more effectively than all other existing models in the detection of fake data.

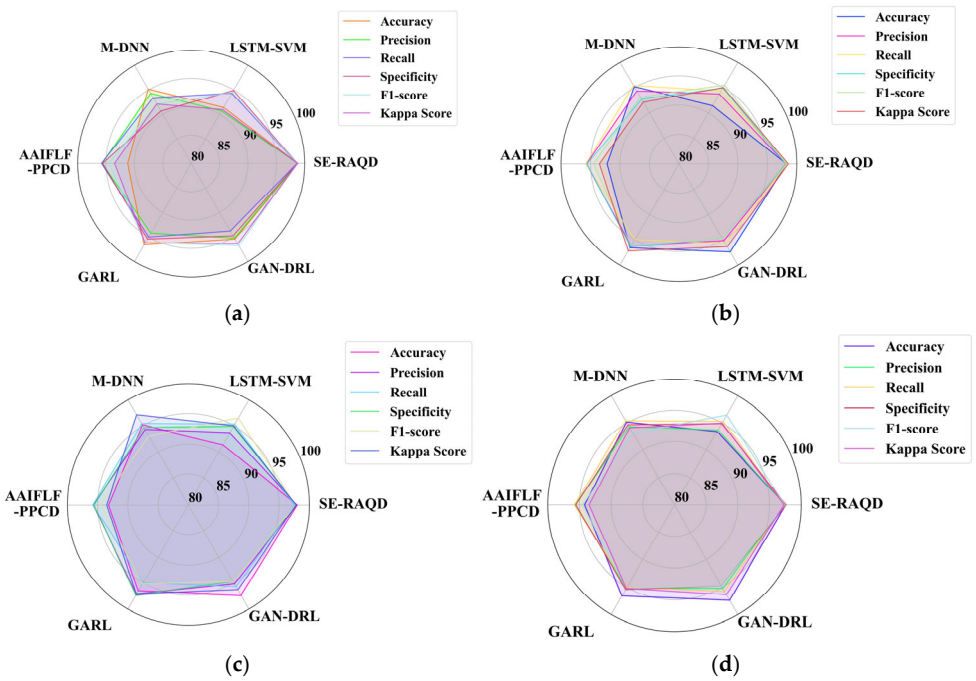


Figure 10. Comparative analysis between the proposed SE-RAQD and existing models on all datasets: (**a**) UNSW-NB15 dataset, (**b**) CICIDS2017 dataset, (**c**) CIC IoT Dataset 2023, and (**d**) NF-ToN-IoT dataset.

Figure 11a–c illustrates the results of a throughput, latency, and CPU utilization analysis of different existing models, including the LSTM-SVM, M-DNN, AAIFLF-PPCD, GARL, GAN-DRL, and proposed SE-RAQD models, on four benchmark datasets. Figure 11a highlights that the SE-RAQD model achieved higher performance than all the other models, with a throughput of 980 (instances/s), 950 (instances/s), 890 (instances/s), and 920 (instances/s) on the UNSW-NB15, CICIDS2017, CIC IoT Dataset 2023, and NF-ToN-IoT datasets, respectively. Figure 11b illustrates the latency analysis for the proposed SE-RAQD model and other existing models. This graph demonstrates that the SE-RAQD model attained lower latencies of 16, 18, 22, and 23 ms on the UNSW-NB15, CICIDS2017, CIC IoT Dataset 2023, and NF-ToN-IoT datasets, respectively. Figure 11c depicts the CPU utilization analysis of the proposed SE-RAQD model and other existing models. This graphical representation highlights that the SE-RAQD model achieves an effective CPU utilization of 18, 22, 25, and 27% on the UNSW-NB15, CICIDS2017, CIC IoT Dataset 2023, and NF-ToN-IoT datasets, respectively.

5.6. Ablation Study

Table 3 highlights the contribution of various components in the SE-RAQD model to detecting fake data. Notably, the performance of a model drops when the feature extraction model, ET-PBO, is eliminated. The full SE-RAQD model, which includes pre-processing and feature extraction approaches, achieved excellent results, with an accuracy of 98.14%, highlighting the combined impact of these components.

Table 3. Ablation result analysis.

Variant	Accuracy	Precision	Recall	Specificity	F1-Score
With preprocessed data	97.56%	97.12%	96.08%	97.10%	96.59%
Without preprocessed data	92.34%	91.76%	90.43%	91.57%	91.09%
Without ET-PBO	95.67%	94.80%	95.12%	94.90%	94.96%
Without meta-ensemble	93.05%	94.10%	93.90%	93.20%	94.00%
Full SE-RAQD Model	98.14%	98.08%	97.99%	98.05%	98.03%

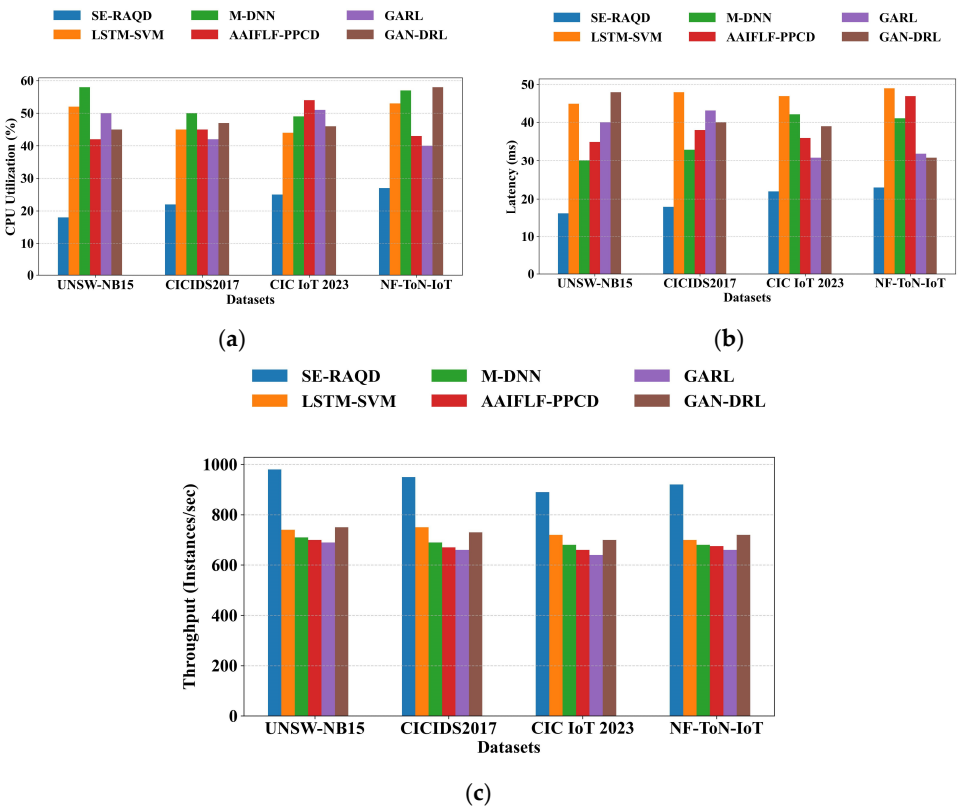


Figure 11. Analysis comparing the proposed SE-RAQD and existing models on all datasets: (**a**) throughput analysis, (**b**) latency analysis, and (**c**) CPU utilization analysis.

5.7. Statistical Analysis

Table 4 presents a statistical analysis using the Wilcoxon signed-rank test to compare the proposed SE-RAQD model with various existing models. Each entry indicates the significance of the performance difference, with ρ-values less than 0.05 for all comparisons indicating that the SE-RAQD model performs significantly better than other models. Consistently low ρ values reinforce the robustness of SE-RAQD and its performance in different contexts, making a compelling argument for its adoption in future applications. Table 4 presents a statistical analysis using the Wilcoxon signed-rank test to compare the proposed SE-RAQD model with various existing models. This test is a non-parametric alternative to the paired *t*-test, suitable for evaluating paired, non-normally distributed performance results. For each pairwise comparison, the SE-RAQD model was evaluated against another baseline model across four benchmark datasets and seven metrics, yielding n = 28 paired observations per comparison. A significance level of $\rho < 0.05$ is used to determine whether the differences were statistically significant. Each entry indicates the significance of the performance difference, with ρ values less than 0.05 for all comparisons, indicating that the SE-RAQD model performs significantly better than the other models.

Table 4. Wilcoxon signed two-tailed rank test.

SI. No	Pairwise Model Comparison	ρ Value	Significance < 0.05
1	Proposed SE-RAQD versus LSTM-SVM	0.0015	Yes
2	Proposed SE-RAQD versus M-DNN	0.0167	Yes
3	Proposed SE-RAQD versus AAIFLF-PPCD	0.0002	Yes
4	Proposed SE-RAQD versus GARL	0.0006	Yes
5	Proposed SE-RAQD versus GAN-DRL	0.0010	Yes

6. Conclusions and Future Scope

With the increasing digital transformation of smart city infrastructures and IoT-enabled environments, data reliability has become critical for effective service delivery, real-time monitoring, and intelligent decision making. To address the threat posed by fake data, a novel fake data detection model is introduced by integrating a robust feature selection method, ET-PBO, with ensemble learning approaches. The proposed SE-RAQD model enhances classification performance by applying the merits of diverse learners and a meta-classifier. The model demonstrates exceptional performance on benchmark datasets, achieving 98.78% accuracy, 98.75% precision, a recall of 98.66%, a specificity of 98.72%, and F1 score of 98.70%, a kappa score of 98.76%, a throughput of 980 (instances/s), 16 ms latency, and 18% CPU utilization on the UNSW-NB15 dataset, proving its effectiveness in high-volume, noise-prone environments. In the future, the proposed model can be integrated with edge and fog computing, which can help achieve decentralized detection and response. Furthermore, adapting the proposed model to multi-model and heterogeneous data can improve its applicability in various smart city domains. In future work, integrating blockchain systems such as Vfchain can further enhance the security and trustworthiness of the fake data detection framework. This is especially valuable in federated learning settings, where data and models are distributed across multiple devices or organizations. By leveraging blockchain, each data transaction or model contribution can be immutably recorded and verified, preventing tampering and enabling traceability. This creates a fully decentralized, privacy-preserving, and auditable system suitable for real-time deployment in smart city and IoT infrastructures.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are openly available at [https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15], [https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset], [https://www.kaggle.com/datasets/akashdogra/cic-iot-2023], and [https://www.kaggle.com/datasets/dhoogla/nftoniot] accessed on 23 June 2025.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Rehan, H. Internet of Things (IoT) in smart cities: Enhancing urban living through technology. *J. Eng. Technol.* **2023**, *5*, 1–16.
2. Mahmood, H.S.; Abdulqader, D.M.; Abdullah, R.M.; Rasheed, H.; Ismael, Z.N.R.; Sami, T.M.G. Conducting In-Depth Analysis of AI, IoT, Web Technology, Cloud Computing, and Enterprise Systems Integration for Enhancing Data Security and Governance to Promote Sustainable Business Practices. *J. Inf. Technol. Inform.* **2024**, *3*, 297–332.
3. Mehta, A.A.; Padaria, A.A.; Bavisi, D.J.; Ukani, V.; Thakkar, P.; Geddam, R.; Kotecha, K.; Abraham, A. Securing the future: A comprehensive review of security challenges and solutions in advanced driver assistance systems. *IEEE Access* **2023**, *12*, 643–678. [CrossRef]
4. Fadhel, M.A.; Duhaim, A.M.; Saihood, A.; Sewify, A.; Al-Hamadani, M.N.; Albahri, A.S.; Alzubaidi, L.; Gupta, A.; Mirjalili, S.; Gu, Y. Comprehensive systematic review of information fusion methods in smart cities and urban environments. *Inf. Fusion* **2024**, *107*, 102317. [CrossRef]
5. Wu, P.; Zhang, Z.; Peng, X.; Wang, R. Deep learning solutions for smart city challenges in urban development. *Sci. Rep.* **2024**, *14*, 5176. [CrossRef]
6. Goumopoulos, C. Smart city middleware: A survey and a conceptual framework. *IEEE Access* **2024**, *12*, 4015–4047. [CrossRef]
7. Mancy, H.; Naith, Q.H. SwinIoT: A hierarchical transformer-based framework for behavioral anomaly detection in IoT-Driven smart cities. *IEEE Access* **2025**, *13*, 48758–48774. [CrossRef]
8. Li, X.; Zhang, D.; Zheng, Y.; Hong, W.; Wang, W.; Xia, J.; Lv, Z. Evolutionary computation-based machine learning for smart city high-dimensional big data analytics. *Appl. Soft Comput.* **2023**, *133*, 109955. [CrossRef]
9. Ahmed, Y.; Beyioku, K.; Yousefi, M. Securing smart cities through machine learning: A honeypot-driven approach to attack detection in Internet of Things ecosystems. *IET Smart Cities* **2024**, *6*, 180–198. [CrossRef]

10. Jogarao, M.; Lakshmanna, B.C.; Naidu, S.T. Ai-enabled circular economy management for sustainable smart cities: Integrating artificial intelligence in resource optimization and waste reduction. In *Smart Cities and Circular Economy*; Emerald Publishing Limited: Leeds, UK, 2024; pp. 83–96.
11. Shabbir, A.; Cheema, A.N.; Ullah, I.; Almanjahie, I.M.; Alshahrani, F. Smart city traffic management: Acoustic-based vehicle detection using stacking-based ensemble deep learning approach. *IEEE Access* **2024**, *12*, 35947–35956. [CrossRef]
12. Asavisanu, N.; Khezresmaeilzadeh, T.; Sequeira, R.; Qiu, H.; Ahmad, F.; Psounis, K.; Govindan, R. CATS: A Framework for Cooperative Autonomy Trust & Security. *IEEE Trans. Veh. Technol.* **2025**, *74*, 10092–10108.
13. Sani, M.S.; Iranmanesh, S.; Salarian, H.; Raad, R.; Jamalipour, A. Bids: Blockchain-enabled intrusion detection system in smart cities. *IEEE Internet Things Mag.* **2024**, *7*, 107–113. [CrossRef]
14. He, D.; Kim, J.; Shi, H.; Ruan, B. Autonomous anomaly detection on traffic flow time series with reinforcement learning. *Transp. Res. Part C Emerg. Technol.* **2023**, *150*, 104089. [CrossRef]
15. Nayak, R.P.; Sethi, S.; Bhoi, S.K.; Sahoo, K.S.; Nayyar, A. Ml-mds: Machine learning based misbehavior detection system for cognitive software-defined multimedia vanets (csdmv) in smart cities. *Multimed. Tools Appl.* **2023**, *82*, 3931–3951. [CrossRef]
16. Ji, C.; Niu, Y. A hybrid evolutionary and machine learning approach for smart city planning: Digital twin approach. *Sustain. Energy Technol. Assess.* **2024**, *64*, 103650. [CrossRef]
17. Saleem, M.; Abbas, S.; Ghazal, T.M.; Khan, M.A.; Sahawneh, N.; Ahmad, M. Smart cities: Fusion-based intelligent traffic congestion control system for vehicular networks using machine learning techniques. *Egypt. Inform. J.* **2022**, *23*, 417–426. [CrossRef]
18. AlJamal, M.; Mughaid, A.; Bani-Salameh, H.; Alzubi, S.; Abualigah, L. Optimizing risk mitigation: A simulation-based model for detecting fake IoT clients in smart city environments. *Sustain. Comput. Inform. Syst.* **2024**, *43*, 101019. [CrossRef]
19. Ajao, L.A.; Apeh, S.T. Secure edge computing vulnerabilities in smart cities sustainability using petri net and genetic algorithm-based reinforcement learning. *Intell. Syst. Appl.* **2023**, *18*, 200216. [CrossRef]
20. Ragab, M.; Ashary, E.B.; Alghamdi, B.M.; Aboalela, R.; Alsaadi, N.; Maghrabi, L.A.; Allehaibi, K.H. Advanced artificial intelligence with federated learning framework for privacy-preserving cyberthreat detection in IoT-assisted sustainable smart cities. *Sci. Rep.* **2025**, *15*, 4470. [CrossRef]
21. Mishra, S.; Chaurasiya, V.K. Hybrid deep learning algorithm for smart cities security enhancement through blockchain and internet of things. *Multimed. Tools Appl.* **2024**, *83*, 22609–22637. [CrossRef]
22. Kumar, A.; Kumar, S. An advance encryption and attack detection framework for securing smart cities data in blockchain using deep learning approach. *Wirel. Pers. Commun.* **2024**, *135*, 1329–1362. [CrossRef]
23. Arif, R.M.; Aslam, M.; Al-Otaibi, S.; Martinez-Enriquez, A.M.; Saba, T.; Bahaj, S.A.; Rehman, A. A deep reinforcement learning framework to evade black-box machine learning based IoT malware detectors using GAN-generated influential features. *IEEE Access* **2023**, *11*, 133717–133729. [CrossRef]
24. Hasan, R.; Biswas, B.; Samiun, M.; Saleh, M.A.; Prabha, M.; Akter, J.; Joya, F.H.; Abdullah, M. Enhancing malware detection with feature selection and scaling techniques using machine learning models. *Sci. Rep.* **2025**, *15*, 9122. [CrossRef] [PubMed]
25. Tahir, M.; Abdullah, A.; Udzir, N.I.; Kasmiran, K.A. A novel approach for handling missing data to enhance network intrusion detection system. *Cyber Secur. Appl.* **2025**, *3*, 100063. [CrossRef]
26. Das, V.; Nair, B.B.; Thiruvengadathan, R. A Novel Feature Encoding Scheme for Machine Learning Based Malware Detection Systems. *IEEE Access* **2024**, *12*, 91187–91216. [CrossRef]
27. Bhatnagar, M.; Rozinaj, G.; Vargic, R. Using crafted features and polar bear optimization algorithm for short-term electric load forecast system. *Energy AI* **2025**, *19*, 100470. [CrossRef]
28. AlShahrani, B.M.M.; Quasim, M.T. Classification of cyber-attack using Adaboost regression classifier and securing the network. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 1215–1223.
29. Zhou, X.; Chen, W.; Li, Y. netQDA: Local Network-Guided High-Dimensional Quadratic Discriminant Analysis. *Mathematics* **2024**, *12*, 3823. [CrossRef]
30. Mhamdi, L.; Isa, M.M. Securing SDN: Hybrid autoencoder-random forest for intrusion detection and attack mitigation. *J. Netw. Comput. Appl.* **2024**, *225*, 103868. [CrossRef]
31. Alserhani, F.; Aljared, A. Evaluating ensemble learning mechanisms for predicting advanced cyber attacks. *Appl. Sci.* **2023**, *13*, 13310. [CrossRef]