International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

ROBUST AND LIGHTWEIGHT CONTRASTIVE RIDGE FEATURE SELECTION FOR EFFICIENT IOT MALWARE DETECTION

Jasmine Jolly[1,1], Ananthanarayanan V[b,1]

[1]Department of Computer Science and Engineering, Amrita School of Computing, Coimbatore 641112, Amrita Vishwa Vidyapeetham, India

Abstract

The connected smart devices, sensors, and systems in the Internet of Things (IoT) enables flawless data exchange and supports automation in healthcare, industry, and smart cities. Though the life gets convenient with increasingly connected world , it also introduces security risks, hence strong protection is essential. The resource constrained IoT devices demands a lightweight malware detection ensuring balance between detection efficacy and computational cost. The selection of appropriate features is required in this regard. A novel method Contrastive Ridge Feature Selection (CRFS) is proposed in this paper , which make use of Ridge regression coefficients to identify the most discriminative features distinguishing between malicious and benign data. The feature ranking of CFRS is according to their contrastive importance between benign and attack classes. We conducted an extensive evaluation of CRFS on IoT malware datasets such as N-BaIoT, Bot-IoT, KDD Cup 2018, and Edge-IIoTset, which demonstrates its effectiveness.The CRFS-based feature selection not only boosted the accuracy of the Linear SVC model—achieving 97.15% on Bot-IoT, 94.85% on Edge-IIoTset, 93.69% on KDD Cup, and 85.06% on N-BaIoT—but also significantly reduced training time by up to 85.84% and minimized feature redundancy to as low as 0.71%, making it highly suitable for real-time IoT malware detection.

Keywords Lightweight IDSs, IoT, Network security,

Feature selection, Machine Learning

1 Introduction

The development of IoT technology has revolutionized industries such as healthcare, industrial automation, and smart cities. According to [1], it is estimated that 64 billion IoT devices are currently in use, leading to an increase in security threats. These threats, particularly the growing risk of malware attacks, pose significant challenges in resourcelimited IoT environments [2]. Such environments are highly vulnerable to cyberattacks, including botnets, ransomware, and DDoS attacks [3]. As a result, there is an urgent need for lightweight and efficient security measures to protect IoT systems against these emerging threats.

Received: July 18, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Existing machine learning models for IoT malware detection commonly utilize system logs, API calls, and network traffic to differentiate between malicious and benign behavior. However, the high dimensionality of features often makes the detection process computationally intensive. Therefore, reducing the number of features without compromising detection accuracy is crucial. Feature selection plays a vital role in enhancing model performance by eliminating redundant or irrelevant features and retaining only the most significant ones for efficient malware detection. Several feature selection techniques, such as chi-square, information gain, and correlation-based methods, have been applied to IoT intrusion detection tasks [4]. In [5], these techniques were analyzed to identify and rank important features. The findings revealed that while adding features initially improves accuracy, the improvement plateaus beyond a certain point. Hence, selecting a meaningful subset of features helps in maintaining competitive accuracy while reducing computational costs, which is essential for resourceconstrained IoT environments.

A hybrid feature selection strategy combining information gain and recursive feature elimination was proposed in [6] and integrated into a cascaded long short-term memory (LSTM) model. This approach demonstrated strong attack classification performance, reinforcing the importance

of robust feature selection in enhancing intrusion detection systems. On the other hand, Federated Learning (FL)-based methods [7] are not well-suited for lightweight IoT malware detection due to their high communication overhead and computational demands. The continuous exchange between edge devices and the central server may introduce latency and increase power consumption, making these approaches impractical for real-time IoT environments. The study in [8] combined Decision Trees with Recursive Feature Elimination to eliminate redundant data and select relevant features. However, this method may encounter challenges related to computational complexity, particularly due to the heavy processing requirements of the Stacking algorithm and the recursive elimination process when applied to large datasets.

Traditional feature selection techniques often depend heavily on specific classifiers and may fail to capture non-linear relationships between features, resulting in inconsistent model performance. Moreover, conventional methods are not scalable for large IoT datasets as they demand substantial computational resources. These limitations emphasize the need for a feature selection method that balances high detection accuracy with low computational cost.

This work proposes Contrastive Ridge Feature Selection (CRFS), a novel approach that leverages Ridge regression coefficients to identify the most discriminative features that differentiate malicious from benign traffic. By incorporating contrastive learning principles, CRFS not only selects suitable features but also effectively captures both linear and non-linear relationships, offering an advantage over traditional methods. To validate its effectiveness, CRFS is compared with existing state-of-the-art feature selection techniques in terms of feature selection time, training time, and classification accuracy. The evaluation is conducted on four

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

widely used IoT malware datasets: N-BaIoT [9], Bot-IoT [10], KDD Cup [11], and Edge-IIoTset [12].

The key contributions of this study are as follows:

– Presenting CRFS, a novel feature selection technique that optimizes feature selection for IoT malware detection by utilizing Ridge regression coefficients.

– Carrying out comprehensive tests to evaluate the detection performance and computing efficiency of CRFS in comparison to current feature selection methods.

– Proving that CRFS is a viable option for real-time IoT security applications by providing an efficient trade-off between accuracy and efficiency.

The remainder of this paper is structured as follows: Section 2 presents related work in IoT malware detection and feature selection techniques. Section 3 describes the proposed CRFS methodology in detail. Section 4 outlines the experimental setup and datasets used. Section 5 presents the results and discussion, while Section 6 concludes the study with future research directions.

2 Related Work

Malware poses a significant security threat to the IoT, with the detection of unknown malware emerging as a critical challenge due to the inherent limitations of IoT devices, such as low computational power and energy constraints, as well as the increased attack surface introduced by new network connectivity methods like cloud services and the integration of non-traditional devices such as smart sensors [13].The rapid growth of resource-constrained IoT devices, along with the rise of Machine-to-Machine (M2M) communication, has significantly increased cybersecurity risks—especially malware threats—highlighting the need for lightweight and efficient deep learning models that can perform real-time malware detection while maintaining high accuracy and low computational overhead.

Traditional malware detection techniques rely on signature based, heuristic based, and behavior-based methods. Signature based approaches, such as those used in antivirus software, compare files against known malware signatures but fail against zero day threats and polymorphic malware [14]. Heuristic based methods analyze code structures and execution patterns to detect anomalies, but they often suffer from high false positive rates [15]. Behavior-based detection monitors runtime activities to identify malicious behaviors, offering better generalization than signature-based methods but incurring significant computational overhead [16].While these techniques have been widely used, their limitations in handling dynamic and evasive malware highlight the need for more robust and lightweight detection approaches in IoT environments.

Machine learning(ML) based detection of malware has gained significant attention in addressing these limitations. ML models, including decision trees, support vector machines (SVM), random forests, and deep learning models, have been utilized to classify network traffic and system logs based on extracted features [17].These approaches have demonstrated higher

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

detection accuracy and generalization capabilities compared to traditional methods. However, highdimensional feature sets in IoT malware datasets often lead to increased computational complexity and overfitting, necessitating effective feature selection techniques to improve model efficiency and interpretability.

Feature selection plays a critical role in IoT malware detection, as it reduces the dimensionality of the data and improves the efficiency of machine learning models. Traditional feature selection methods, such as filter-based approaches (e.g., mutual information) and wrapper-based techniques (e.g., recursive feature elimination), have been widely used in malware detection [18] . However, high-dimensional data from modern technologies often challenge the interpretability and stability of traditional feature selection methods, as correlated features can lead to multiple equally optimal selections, reducing confidence in the chosen feature subset [19]. Computational constraints often make wrapper and embedded methods impractical for large datasets, despite their ability to model feature dependencies and enhance classifier accuracy compared to filter methods so that it is not suitable for lightweight IoT applications [20]. Metaheuristic based approaches, such as the Firefly Algorithm, Genetic Algorithm, and Particle Swarm Optimization (PSO), have been employed to optimize feature selection by identifying the most relevant subset of features [21]. However, these evolutionary techniques require multiple iterations to converge, making them less suitable for real-time IoT applications where rapid decision-making is essential.Recently, ridge regression-based feature selection has gained attention due to its ability to handle multicollinearity and provide stable feature importance scores [22]. The studies in [23–25] highlights the growing shift toward memory analysis, generative models, and deep learning in addressing the limitations of traditional malware detection techniques. These methods are unsuitable for lightweight IoT malware detection due to their high computational and memory demands, which exceed the processing capabilities of resource-constrained IoT devices. Additionally, deep learning and GAN-based approaches introduce latency and network overhead, making them impractical for real-time IoT environments. [26] have explored the use of Markov images with CNNs, while effective for obfuscated malware detection, may still pose challenges in lightweight IoT environments due to the computational complexity of deep learning models.

Ridge regression assigns regularized coefficients to features, effectively reducing overfitting while retaining important predictive attributes. However, traditional ridge-based methods do not explicitly enhance feature separability, limiting their effectiveness in malware detection scenarios where distinguishing between similar classes is crucial. To further improve feature selection efficiency, contrastive learning, a self supervised learning approach, has been explored in domains such as computer vision and natural language processing. Contrastive learning enhances feature representations by maximizing the similarity between positive samples (similar examples) while minimizing the similarity between negative samples (dissimilar examples) [27].Though contrastive learning has demonstrated promising results in deep learning applications, its integration with feature selection for IoT malware detection remains largely unexplored.

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

The Contrastive Ridge Feature Selection (CRFS) proposed in this paper aims to bridge this gap by leveraging ridge regression coefficients in combination with contrastive learning principles to enhance feature selection. By integrating contrastive learning, the proposed method ensures that selected features maximize class separability while maintaining computational efficiency. This approach enables the selection of a minimal yet highly discriminative feature subset, making it suitable for lightweight and real-time IoT malware detection.

Problem Formulation

Let $X \in R^{N \times d}$ *represent a dataset with N samples and d features. Each sample* $x_i \in R^d$ *is associated with a class label* $y_i \in \{0,1\}$*, where:*

- $y_i = 0$ *denotes a benign sample.*
- $y_i = 1$ *denotes an attack sample.*

The goal of Contrastive Ridge Feature Selection (CRFS) is to identify the subset of features that exhibit the most discriminative power between benign and attack classes while ensuring computational efficiency.

3 Methodology

The feature selection using Ridge regression implemented in [22] is known as importance based method involves selecting features whose absolute coefficients exceed the average magnitude of all coefficients. However, this method has several drawbacks: The use of a global average coefficient threshold treats all features as equally important across classes. This approach ignores features that might be highly informative for distinguishing classes but have absolute coefficients close to the average.Ridge regression applies L2 regularization, which shrinks feature coefficients towards zero, especially in high-dimensional datasets. As a result, many informative features may be down-weighted below the threshold, leading to suboptimal selection.A single Ridge model does not differentiate between how a feature contributes to benign and attack classes separately. Some features may be informative for one class but irrelevant for the other, making a single global threshold unreliable.

The proposed method of Contrastive Ridge Feature selection shown in Figure 1 overcomes these limitations by leveraging contrastive learning principles applied to Ridge regression. CRFS uses a different approach by training two separate Ridge regression models: one for attacks and one for benign samples, instead of using a single model. This helps capture the differences in feature importance between malicious and normal traffic more effectively.By analyzing the contrast in Ridge regression coefficients between these two models, CRFS identifies the most critical features for detecting IoT malware. The overview of the dataset selection, the feature selection process, the training of machine learning models, and the evaluation strategy used to assess performance are provided in this section.

3.1 Dataset Selection

To evaluate the effectiveness of Contrastive Ridge Feature

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

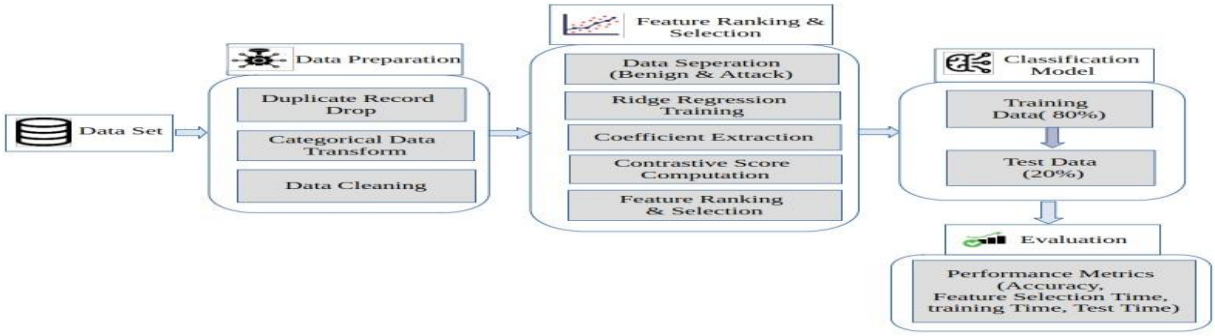


Fig. 1: Feature Selection and Classification Workflow

Selection (CRFS), we use four widely recognized IoT malware datasets. These datasets cover various attack scenarios, device types, and network characteristics, providing a comprehensive test environment to assess the reliability and adaptability of our approach.

– N-BaIoT: This dataset records network activity from IoT devices like thermostats and smart cameras in both normal and botnet-infected states. The devices were intentionally infected with Bashlite and Mirai botnets, providing a realistic view of IoT-based DDoS attacks.

– Bot-IoT: Bot-IoT is a large dataset for IoT security research that contains both normal and botnet traffic. It includes various attack types like DDoS, DoS, scanning, keylogging, and data theft, making it useful for intrusion detection studies.

– Edge-IIoTset: This dataset focuses on security threats in Industrial IoT (IIoT), especially in edge-based networks. It includes various attacks like data injection, port scanning, and man-in-the-middle (MITM). Since it targets resource-limited edge environments, it is useful for testing machine learning models in constrained IoT settings.

To ensure the reliability and efficiency of our feature selection and classification process, we applied a structured preprocessing pipeline to these datasets. The key steps in the preprocessing phase are outlined in the following sections.

3.2 Data Preprocessing

A standardized preprocessing pipeline was applied to ensure consistency across the N-BaIoT, BoT-IoT, and KDD Cup datasets. Additionally, dataset-specific modifications were performed for Edge-IIoTset to enhance generalizability.

3.2.1 Common Preprocessing Steps

The following steps were uniformly applied to N-BaIoT, BoT-IoT, and KDD Cup datasets to prepare the data for machine learning models:

– Data Loading & Format Handling: To ensure format compatibility, datasets were loaded from .csv or.xlsx files.

– Elimination of Redundant Features: To get rid of noninformative attributes, columns with only zero values were removed.

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

– Duplicate Record Removal: To guarantee data uniqueness, identical rows were found and eliminated.

– Encoding Categorical Features: To transform categorical features into numerical values appropriate for machine learning models, label encoding was used.

– Target Label Assignment: All other attributes were regarded as predictors, and the "attack" column was identified as the classification target.

3.2.2 Preprocessing for Edge-IIoTset

Some MQTT-based metadata features, like mqtt.protoname, mqtt.topic, mqtt.len, mqtt.proto_len, and mqtt.msgtype, were discovered to produce unreasonably high accuracy (100%) during early trials, indicating data leakage. The model's generalizability was compromised by these variables' significant correlations with attack labels. When they were eliminated, contrastive score-based selection stabilized the accuracy of the model at 94.85%, indicating their deceptive influence.Additional non-essential metadata features were removed from the Edge-IIoTset in order to enhance generalization.These preprocessing procedures ensured robustness in feature selection and model evaluation by standardizing the datasets.

3.3 Proposed Contrastive Ridge Feature Selection (CRFS)

The core idea of CRFS is to measure the contrastive importance of features by training separate Ridge regression models on benign and malicious subsets of the dataset. Ridge regression is a linear regression variant that introduces L2 regularization, penalizing large coefficients to prevent overfitting. The Ridge model solves the following optimization problem:

$$N\beta = \text{argmin} \sum (y_i - X_i\beta)^2 + \alpha\|\beta\|_2^2\beta \, i{=}1$$

where α is the regularization parameter controlling the tradeoff between model complexity and fit. The Ridge coefficients β reflect the importance of each feature in predicting the target variable. The proposed method leverages Ridge regression models trained separately on benign and attack samples to derive a contrastive feature importance score. The process consists of the following steps:

1. Data Separation: Given a dataset X with corresponding labels y, the feature matrix is split into benign samples

(X_{benign}, where $y = 0$) and attack samples (X_{attack}, where $y = 1$).

2. Ridge Regression Training: Two Ridge regression models are trained independently:

– A Ridge model is fitted on X_{benign} using a dummy output vector of zeros.

– Another Ridge model is trained on X_{attack} using a dummy output vector of ones.

3. Coefficient Extraction: The learned coefficients from both models, β_{benign} and β_{attack}, represent the influence of each feature in distinguishing between benign and attack classes.

4. Contrastive Score Computation: A contrastive score for each feature is calculated as:

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$S_i = \frac{|\beta_{\text{attack},i} - \beta_{\text{benign},i}|}{\text{median}(\beta_{\text{attack}} \cup \beta_{\text{benign}}) + \varepsilon} \quad (1)$$

where ε is a small value to prevent division by zero.

5. Feature Ranking and Selection: Features are ranked based on their contrastive scores, and the top features are selected as the most discriminative ones.

Unlike standard coefficient-based selection, which may retain redundant features with moderate importance across both classes, CRFS prioritizes features that exhibit the greatest differentiation between benign and attack behaviors. Since CRFS uses relative differences between two Ridge models, it is less affected by Ridge's tendency to shrink coefficients, ensuring that critical features are not eliminated due to overregularization. Traditional methods rely on fixed thresholds, which may not generalize across different datasets. CRFS dynamically selects features based on dataset-specific class differences, leading to more adaptable and effective feature selection.

Computational Complexity Let n_0 represent the number of samples in the benign class and n_1 be the number of samples in the attack class. The Ridge regression has a closed-form solution complexity of $O(d^2 \min(n,d))$. The total complexity of CRFS, which trains two distinct Ridge models, is:

$$O(d^2 \min(n_0,d)) + O(d^2 \min(n_1,d)) \quad (2)$$

which is still more efficient than sequential forward selection techniques that necessitate iterative model retraining.

4 Classification Models

Linear Support Vector Classification (Linear SVC) is employed as our primary classifier to evaluate the impact of our proposed Contrastive Ridge Feature Selection (CRFS) method on model performance, particularly for IoT malware detection. The choice of Linear SVC is driven by its ability to efficiently handle high-dimensional data while maintaining low computational complexity, which is crucial for resource-constrained IoT environments[22]. Linear SVC is particularly suited for such environments because it scales well with large datasets and maintains a relatively low memory footprint, ensuring faster training and inference times. Another key advantage of Linear SVC is its direct feature weight coefficients, which enhance the interpretability of the selected features. Unlike non-linear SVMs, which rely on complex kernel functions, Linear SVC enables straightforward identification of the most influential features, helping to refine the dataset and improve model accuracy.

IoT malware datasets typically include a large number of redundant and irrelevant features, which can reduce classification accuracy and increase training and testing times. CRFS effectively addresses this challenge by selecting the most discriminative features, leading to improved accuracy, reduced overfitting, and faster model execution. The combination of Linear SVC with CRFS thus results in a more computationally efficient and interpretable

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

model, making it highly suitable for real-world IoT malware detection scenarios where rapid and accurate classification is essential.

5 Result Analysis

To validate the effectiveness of our proposed feature selection method, we conducted experiments on four widely used IoT security datasets: N-BaIoT, Bot-IoT, KDD Cup 99, and Edge-IIoTset. These datasets include both benign and malicious network traffic, covering various attack types. Each dataset contains a large number of features, from which we apply our proposed method to identify the most relevant features for lightweight IoT malware detection.

Table 1 provides an overview of the feature sets available in each dataset used in this study, including N-BaIoT, BotIoT, KDD Cup 99, and Edge-IIoTset. These datasets contain a variety of network traffic and system behavior features, which serve as the foundation for IoT malware detection.Table 2 presents the features selected using the proposed contrastive feature selection method. These features were chosen based on their significance in distinguishing normal and malicious activities while ensuring computational efficiency, making them suitable for lightweight IoT environments. The selected features vary across datasets, focusing on statistical properties, network behavior, and protocol specific characteristics.

Distinct and well-separated distributions shown in Figures 2, 3, 4 and 5 indicate that a feature is highly relevant for classification, while overlapping distributions suggest lower discriminative power. In our analysis, features selected using the contrastive score method demonstrated a clear distinction between normal and attack traffic, reinforcing their importance in classification. We evaluate the impact of Contrastive Ridge Feature Selection (CRFS) on classification accuracy and computational efficiency using Linear SVC across four IoT malware datasets. The results demonstrate that using CRFS-selected features not only improves accuracy but also significantly reduces feature selection and training time compared to using the full feature set.

Tables 3, 4, 5 and 6 present the performance metrics of the Contrastive Ridge Linear SVC model on the N-BaIoT, KDD Cup, Bot-IoT, and Edge-IIoT datasets, evaluating precision, recall, F1-score, and accuracy. On the N-BaIoT dataset (Table 3), the CRFS based Linear SVC model achieved an accuracy of 0.8506 with a macro F1-score of 0.8069 . For the KDD Cup dataset (Table 4), the model showed high accuracy (0.9369) and a macro F1-score of 0.9325. The Bot-IoT dataset results (Table 5) showed excellent accuracy (0.9715) with a macro F1-score of 0.9278. On the Edge-IIoT dataset (Table 6), the model achieved an accuracy of 0.9485 and a macro F1-score of 0.8862. The results confirm that the CRFS-based feature selection significantly improves classification accuracy and reduces computational cost across diverse IoT datasets.Features with well-separated distributions led to higher precision and recall, particularly in contrastive score-based selection.

Table 7 presents the accuracy of Linear SVC when trained on the full feature set and feature subsets selected using different selection methods, including CRFS, importancebased

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

selection, forward feature selection, and Pearson correlation. Notably, CRFS achieves higher accuracy than the full feature set in all datasets. The results demonstrate that CRFS achieves higher accuracy than the other methods in most datasets. For instance, in the Bot-IoT dataset,accuracy improves drastically from 11.31% to 97.15% with CRFS which is significantly higher than importance-based selection (91.62%) and forward feature selection (92.06%). Similarly, in the KDD Cup dataset, CRFS achieves 93.69% accuracy compared to 93.65% for importance-based selection and 92.78% for forward feature selection. Similarly, the accuracy for the Edge-IIoTset dataset increases from 48.58% to 94.85%, demonstrating the effectiveness of CRFS in improving classification performance.

Feature redundancy significantly affects feature selection methods. Data sets with lower redundancy benefited more from Contrastive Ridge Feature Selection (CRFS), while those with higher redundancy showed minimal improvement. For example, in Bot-IoT (1.28% redundancy), CRFS improved performance from 11.31 % to 97.15 %, whereas in N-BaIoT (6.71% redundancy), the improvement was minimal (84.39 % to 85.06 %). This suggests that CRFS works best when redundancy is low. Forward Feature Selection (FFS) was inconsistent, especially in N-BaIoT, where it resulted in a higher time (99%), indicating difficulty in handling redundant features.Table 10 summarizes the redundancy values and their impact on CRFS performance. These results show that lower redundancy allows CRFS to extract more meaningful features, leading to better performance gains.

Table 2: Features Selected Using the Proposed Method

Dataset	Selected Features
N-BaIoT	HpHp_L0.01_weight, HpHp_L0.01_mean, HpHp_L0.01_std, HpHp_L0.01_magnitude, HpHp_L0.01_radius, HpHp_L0.01_covariance, HpHp_L0.01_pcc

Dataset	Feature Categories	Total Features
N-BaIoT	Statistical features, entropy, time-based features	115
Bot-IoT	Network flow features, Protocol Parameters	46

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

KDD Cup 99	Host-based and network-based features	41

Table 1: Available Features in Each Dataset

KDD Cup 99	dst_host_diff_srv_rate, dst_host_same_src_port_rate, dst_host_srv_diff_host_rate, dst_host_serror_rate, dst_host_srv_serror_rate, dst_host_rerror_rate, dst_host_srv_rerror_rate
Edge-IIoTset	dns.qry.name.len, dns.qry.qu, dns.retransmission, dns.retransmit_request, mqtt.conflags, mqtt.hdrflags, mqtt.conflag.cleanses

Table 3: Performance Metrics of Linear SVC on N-BaIoT

	Precision	Re-Call	F1-Score	Support
Normal	0.8203	0.9938	0.8987	45424
Attack	0.9785	0.5633	0.7153	22650
Accuracy			0.8506	68074
Macro AVG	0.8994	0.7786	0.8069	68074

Table 4: Performance Metrics of Linear SVC on KDD Cup

	Precision	Re-Call	F1-Score	Support
Normal	0.9393	0.8925	0.9153	12146
Attack	0.9355	0.9643	0.9497	19644
Accuracy			0.9369	31790
Macro AVG	0.9374	0.9284	0.9325	31790
Weighted AVG	0.9369	0.9369	0.9365	31790

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Weighted AVG	0.8729	0.8506	0.8376	68074

Table 5: Performance Metrics of Linear SVC on Bot IoT

	Precision	Re-Call	F1-Score	Sup
Normal	0.7724	1	0.8716	112
Attack	1	0.9685	0.984	104
Accuracy			0.9715	115
Macro AVG	0.8862	0.9842	0.9278	115
Weighted AVG	0.978	0.9715	0.9731	115

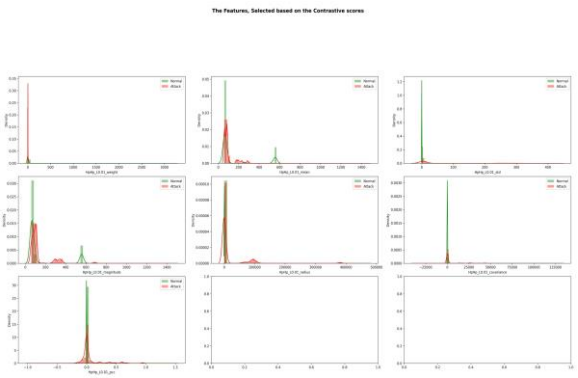
Table 6: Performance Metrics of Linear SVC on Edge IIoT

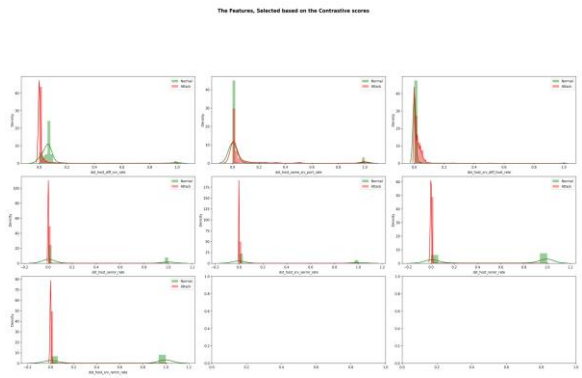
	Precision	Re-Call	F1-Score	Supp
Normal	1	0.6694	0.8019	1005
Attack	0.9425	1	0.9704	5451
Accuracy			0.9485	6456
Macro AVG	0.9712	0.8347	0.8862	6456
Weighted AVG	0.9515	0.9485	0.9442	6456

Table 8 presents the feature selection time comparison across different methods, including Contrastive Ridge Feature Selection (CRFS), importance-based selection, forward feature selection, and Pearson correlation. The results indicate that CRFS consistently requires the least computational time across all datasets. For instance, in the Bot-IoT dataset, Contrastive Feature Selection takes only 0.14 ms, whereas importance-based selection, forward feature selection, and Pearson correlation take 7.82 ms, 822.49 ms, and 58.44 ms, respectively. Similarly, in the N-BaIoT dataset,

CRFS takes 4403.23 ms, which is significantly lower than forward feature selection (67818.31 ms) and Pearson correlation (9471.21 ms).

The training time for Linear SVC using the full set of features and the selected features of CRFS is shown in Table 9. The results indicate that training with the selected features of CRFS significantly reduces training time. For example, in the N-BaIoT dataset, the the training time decreases f15,465.599 to 4,005.04 ms, highlighting the efficiency gain achieved by feature





Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

selection. This substantial reduction in computation time highlights the efficiency of CRFS, making it a suitable choice for real-time applications where quick reFig. 2: Features selected on N-BaIoT Dataset Based on

Contrastive Score

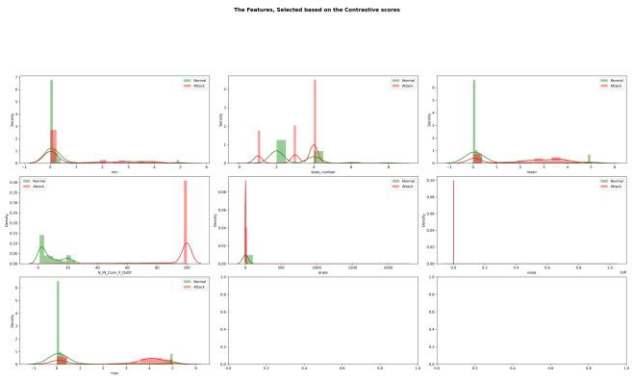


Fig. 3: Features selected on Kdd-Cup Dataset Based on

Contrastive Score

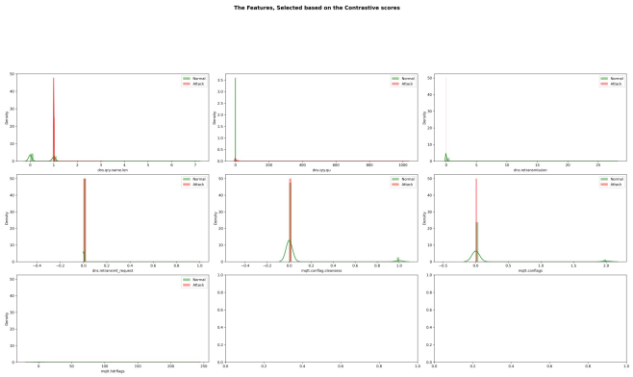


Fig. 4: Features selected on Bot-IoT Dataset Based on

Contrastive Score

Fig. 5: Features selected on Edge-IIoT Dataset Based on

Contrastive Score

sponse times are essential. The reduced selection time with CRFS allows faster model training and prediction, enhancing the overall performance of IoT malware detection systems. The empirical results demonstrate that CRFS achieves higher classification accuracy compared to the traditional Ridge coefficient thresholding method. Across multiple IoT malware detection datasets, CRFS consistently selects features that improve model performance. The observed improvements stem from its ability to enhance class separability while reducing feature redundancy, leading to more efficient and interpretable models.

Figures 6, 7 and 8 analyze the impact of different feature selection methods on IoT malware datasets, including N-BaIoT, Bot-IoT, KDD Cup, and Edge-IoT. Figure 6 shows that while

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

most feature selection methods maintain high accuracy, some, such as the Pearson coefficient in the EdgeIoT dataset, result in slight drops. Figure 7 highlights the time required for feature selection, where methods such as importance-based and forward selection take significantly longer, while CFS is more efficient. Figure 8 demonstrates that the use of selected features reduces the training time compared to the use of all features, with CFS offering a good balance between speed and accuracy. These figures illustrate the trade-off between efficiency and performance, showing that while feature selection speeds up training, careful method selection is needed to maintain accuracy.

6 Discussion

The results emphasize the need to balance computational efficiency and accuracy in IoT malware detection.Forward Feature Selection achieves 99% accuracy on the N-BaIoT dataset but takes 67,818.31 milliseconds, making it impractical for real-time use. In contrast, Correlation-based Feature Selection (CRFS) offers a better balance. On the BotIoT dataset, it reaches 97.15% accuracy with a selection time of just 0.14 milliseconds, making it ideal for resourcelimited IoT environments that require fast processing. Training time analysis further reinforces the advantage of using feature selection methods over full feature sets. For instance, in the KDD Cup dataset, training a model on full features takes 2539.39 ms, whereas using CRFS reduces this time to 701.46 ms without compromising much on accuracy (93.69% vs. 91.34%). Si Table 7: Accuracy Comparison Across Feature Selection Methods

milarly, in the N-BaIoT dataset, training with

Dataset	Model	Full Features(%)	CRFS(%)	Importance Based(%)	Forward Feature Selection(%)	Pearson Coefficient(%)
N-BaIoT	LSVC	84.39	85.06	78.21	99	94.64
Bot-IoT	LSVC	11.31	97.15	91.62	92.06	96.02
KDD Cup	LSVC	91.34	93.69	93.65	92.78	92.88
Edge-IIoTset	LSVC	80.47	94.85	93.99	89.57	63.74

Table 7: Accuracy Comparison Across Feature Selection Methods

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 8: Feature Selection Time Comparison

Dataset	Model	CRFS (ms)	Importance Based (ms)	Forward Feature Selection (ms)	Pearson Coefficient (ms)
N-BaIoT	LSVC	4403.23	9171.65	67818.31	9471.21
Bot-IoT	LSVC	0.14	7.82	822.49	58.44
KDD Cup	LSVC	61.23	97.54	7645.14	9460.517
Edge-IIoTset	LSVC	109.921	286.960	14388.878	16255.407

Table 9: Training Time Comparison

Dataset	Model	Full Features (ms)	CRFS (ms)	Importance based (ms)	Forward feature selection (ms)	Pearson Coefficient (ms)
N-BaIoT	LSVC	15465.590	4309.486	4664.718	4916.343	7080.049
Bot-IoT	LSVC	41.474	35.064	80.869	36.347	74.845
KDD Cup	LSVC	2539.389	771.053	862.992	1152.05	3540.097
Edge-IIoTset	LSVC	7542.878	3568.026	3684.522	4538.469	8678.993

Table 10: Feature Redundancy and CRFS Performance Across Datasets

Dataset	Feature Redundancy (%)	CRFS Performance (%)	Improvement from Full Features (%)
N-BaIoT	6.71	85.06	+0.67
Bot-IoT	1.28	97.15	+85.84
KDD Cup	2.1	93.69	+2.35
Edge IIoTset	0.71	94.85	+14.38

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

full features takes 15,465.59 ms, while CRFS cuts this down to 4005.04 ms, demonstrating its efficiency.

Pearson-based selection shows competitive accuracy but is slightly slower than CRFS in most cases. Importance based selection, while theoretically sound, suffers from lower accuracy on datasets like Edge-IIoTset, where it drops to 46.92%, making it less reliable. Overall, the findings suggest that selecting the optimal method for feature selection depends on the constraints of the deployment environment. Forward Feature Selection can be justified in offline, high-accuracy scenarios, but for real-time applications in IoT malware detection, CRFS emerges as a strong candidate due to its efficiency and competitive performance. The statistical tests, including the Wilcoxon test, indicate no significant differences between most methods in terms of accuracy, further emphasizing the need to prioritize computational cost when deploying these models in real world scenarios.

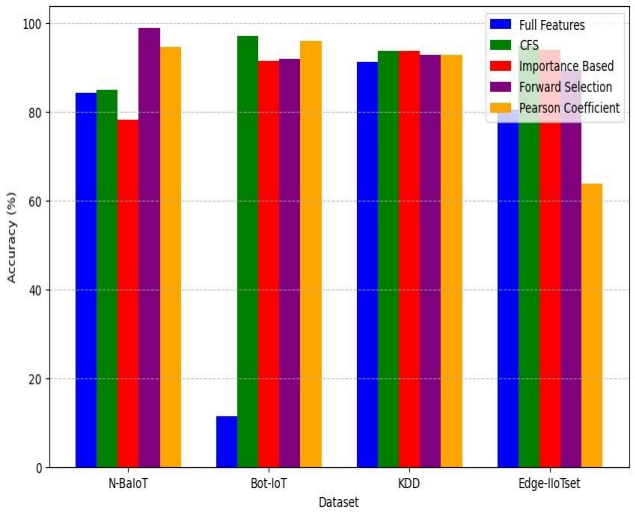


Fig. 6: Accuracy Comparison Across Feature Selection

Methods

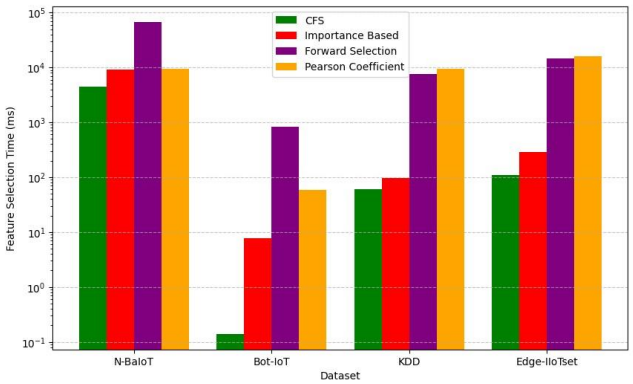


Fig. 7: Feature Selection Time Comparison Across Feature Selection Methods

Received: July 29, 2025

International Journal of Applied Mathematics

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

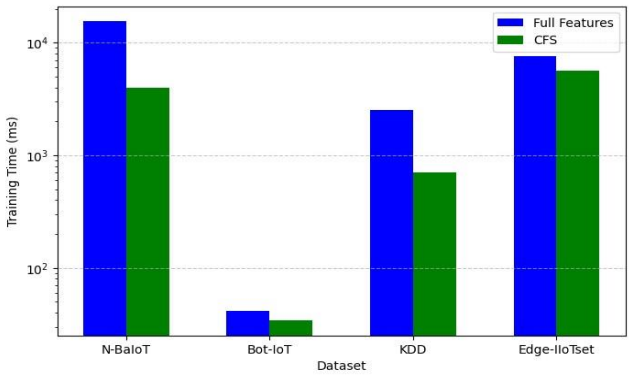


Fig. 8: Training Time Comparison Across Feature Selection Methods

7 Conclusion

The proposed Contrastive Ridge Feature Selection (CRFS) method demonstrates significant improvements over traditional feature selection by evaluating the contrastive importance of features between benign and malicious data. Rather than relying solely on absolute coefficient values from a single model, CRFS leverages separate Ridge regression models to identify features that contribute specifically to distinguishing attack traffic from normal behavior. This contrastive approach enhances class separability, reduces redundancy among selected features, and adapts more effectively across diverse IoT malware datasets. As a result, higher classification accuracy is achieved by CRFS while substantially lowering computational cost, making it a lightweight yet robust solution for real-time IoT malware detection. Future work will focus on integrating CRFS with adaptive machine learning methods to further improve resilience against evolving and dynamic threat patterns.

References

1. Adnan Noor Mian, Syed Waqas Haider Shah, Sanaullah Manzoor, Anwar Said, Kurtis Heimerl, Jon Crowcroft, A value-added IoT service for cellular networks using federated learning,Computer Networks, Volume 213, 2022, 109094, ISSN 1389-1286,https://doi.org/10.1016/j.comnet.2022.109094.
2. Sikder, A. K., Petracca, G., Aksu, H., Jaeger, T., & Uluagac, A. S. (2019):A survey on sensor-based threats and attacks to smart devices and applications., IEEE Communications Surveys & Tutorials, 21(2), 1746-1770.
3. Mosenia, A., & Jha, N. K. (2017). A comprehensive study of security of Internet-of-Things. IEEE Transactions on Emerging Topics in Computing, 5(4), 586-602.
4. Muhammad Shafiq, Zhihong Tian, Ali Kashif Bashir, Xiaojiang Du, Mohsen Guizani, IoT malicious traffic identification using wrapper-based feature selection mechanisms, Computers & Security, Volume 94, 2020, 101863,ISSN 0167-4048, https://doi.org/10.1016/j.cose.2020.101863.

Received: July 29, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

5. Sarhan M., Li, F., & Zhao, J. (2021). Feature selection for IoT intrusion detection using chi-square, information gain, and correlation-based techniques. ArXiv preprint arXiv:2108.12732.
6. Sundaram, Karthic Natarajan, Yuvaraj Perumalsamy, Anitha Ali, Ahmed. (2024). A Novel Hybrid Feature Selection with Cascaded LSTM: Enhancing Security in IoT Networks. Wireless Communications and Mobile Computing. 2024. 1-15. 10.1155/2024/5522431.
7. Pelekoudas-Oikonomou, F., Mirzaee, P. H., Hathal, W., Mantas, G., Rodriguez, J., Cruickshank, H., Sun, Z. (2024). Federated learning-based intrusion detection systems for massive IoT. In G. Mantas, F. Saghezchi, J. Rodriguez, V. Sucasas (Eds.), 6G security and privacy (Chapter 4). Wiley. https://doi.org/10.1002/9781119988007.ch4
8. Lian, Wenjuan, et al. "An Intrusion Detection Method Based on Decision Tree-Recursive Feature Elimination in Ensemble Learning." Mathematical Problems in Engineering 2020.1 (2020): 2835023.
9. Meidan, Y., Bohadana, M., Shabtai, A., Ochoa, M., Tippenhauer, N. O., Elovici, Y. (2018). N-BaIoT: Networkbased detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Computing, 17(3), 47-55.
10. Koroniotis, N., Moustafa, N., Sitnikova, E., Slay, J. (2019). Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. Future Generation Computer Systems, 100, 779-796.
11. Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A. A. (2009). A detailed analysis of the KDD CUP 99 dataset.In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications (pp. 1-6).
12. Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., Janicke, H. (2022). Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning. IEEE Access, 10, 40277-40290.
13. Alrubayyi H, Goteng G, Jaber M, Kelly J. Challenges of Malware Detection in the IoT and a Review of Artificial Immune System Approaches. Journal of Sensor and Actuator Networks. 2021; 10(4):61. https://doi.org/10.3390/jsan10040061
14. Ö. A. Aslan and R. Samet, "A Comprehensive Review on Malware Detection Approaches," in IEEE Access, vol. 8, pp. 6249-6271, 2020, doi: 10.1109/ACCESS.2019.2963724.
15. Vasani, V.; Bairwa, A.K.; Joshi, S.; Pljonkin, A.; Kaur, M.; Amoon, M. Comprehensive Analysis of Advanced Techniques and Vital Tools for Detecting Malware Intrusion. Electronics 2023, 12, 4299. https://doi.org/10.3390/electronics12204299
16. W. Liu, P. Ren, K. Liu and H. -x. Duan, "BehaviorBased Malware Analysis and Detection," 2011 First International Workshop on Complexity and Data Mining, Nanjing, China, 2011, pp. 39-42, doi: 10.1109/IWCDM.2011.17.

Received: July 29, 2025

Volume 38 No. 3s, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

17. Afeez Ajani Afuwape, Ying Xu, Joseph Henry Anajemba, Gautam Srivastava, Performance evaluation of secured network traffic classification using a machine learning approach,Computer Standards Interfaces, Volume 78, 2021, 103545, ISSN 0920-5489, https://doi.org/10.1016/j.csi.2021.103545.

18. Guyon, Isabelle, and André Elisseeff. "An introduction to variable and feature selection." Journal of machine learning research 3.Mar (2003): 1157-1182

19. Utkarsh Mahadeo Khaire, R. Dhanalakshmi,Stability of feature selection algorithm: A review, Journal of King Saud University - Computer and Information Sciences, Volume 34, Issue 4, 2022, Pages 1060-1073, ISSN 13191578, https://doi.org/10.1016/j.jksuci.2019.06.012.

20. Pudjihartono N, Fadason T, Kempa-Liehr AW and O'Sullivan JM (2022) A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction.Front. Bioinform. 2:927312. doi: 10.3389/fbinf.2022.927312

21. Nssibi, Maha, Ghaith Manita, and Ouajdi Korbaa. "Advances in nature-inspired metaheuristic optimization for feature selection problem: A comprehensive survey." Computer Science Review 49 (2023): 100559.

22. Jahongir Azimjonov, Taehong Kim, Designing accurate lightweight intrusion detection systems for IoT networks using fine-tuned linear SVM and feature selectors, Computers Security, Volume 137, 2024, 103598, ISSN 01674048,https://doi.org/10.1016/j.cose.2023.103598.

23. P. M. Krishna, D. B. Kumar and K. Ashwini, "Malware Detection in Image Data: A Deep Learning Approach," 2024 5th International Conference on Communication, Computing & Industry 6.0 (C2I6), Bengaluru, India, 2024, pp. 1-5, doi: 10.1109/C2I663243.2024.10894877.

24. K. M. Balasubramanian et al., "Obfuscated Malware detection using Machine Learning models," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-8, doi: 10.1109/ICCCNT56998.2023.10307598.

25. V. Amrith et al., "An early malware threat detection model using Conditional Tabular Generative Adversarial Network," 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), Delhi, India, 2023, pp. 1-8, doi: 10.1109/ICCCNT56998.2023.10307903.

26. D. K. A. et al., "Obfuscated Malware Detection in IoT Android Applications Using Markov Images and CNN," in IEEE Systems Journal, vol. 17, no. 2, pp. 2756-2766, June 2023, doi: 10.1109/JSYST.2023.3238678.

27. Kumar, P., Rawat, P. Chauhan, S. Contrastive selfsupervised learning: review, progress, challenges and future research directions. Int J Multimed Info Retr 11, 461–488 (2022). https://doi.org/10.1007/s13735- 022-00245-6

Received: July 29, 2025