

Correlation-Aware Hybrid Feature Selection for Efficient Machine Learning-Based Intrusion Detection in Smart Home IoT Networks

Samar M. Alqhtani 

Department of Information Systems, College of Computer Science and Information Systems,
Najran University, Najran, 61441, Saudi Arabia

Abstract—With the emergence of smart homes based on the Internet of Things, network-based attacks have become more prevalent and require effective and lightweight intrusion detection (ID). In this study, a Correlation-Aware Hybrid Feature Selection (CAHFS) framework for smart home IDS is proposed. The proposed approach fuses correlation analysis and Mutual Information (MI) and Random Forest (RF) feature importance to eliminate redundant and less important features and keep only those features that contribute most to the classification. To evaluate the different feature-selection strategies, four machine-learning models were tested: Logistic Regression (LR), Decision Tree (DT), Random Forest (RF) and HistGradientBoosting. Among the classifiers, HistGradientBoosting provides the most stable results out of the assessed models. The proposed CAHFS feature set, using only 16 features, yielded 91% accuracy, 91% precision, 90% recall, 90% F1-score, and a ROC-AUC of 0.96. It was also able to attain an accuracy level of about 95% specificity with an overall error rate of 8.9%, which provides a good balance between detecting the attacks and avoiding false alarms. The comparison with other feature-selection methods demonstrated that CAHFS managed to decrease the number of features from 22 to 16, with the same or similar classification results. Overall, the findings indicate that CAHFS can offer a more compact and informative feature set to be used in machine-learning-based intrusion detection in smart-home networks, which may benefit networks with limited computational resources.

Keywords—Smart home; intrusion detection; machine learning-feature representation; correlation-aware hybrid feature selection

I. INTRODUCTION

The Internet of Things (IoT) has transformed the way people today live in their homes by bringing many different types of devices together, from cameras to sensors, smart devices to lighting systems, voice assistants to access-control systems. They often use communication technologies like Wi-Fi, Bluetooth, and Zigbee, and may even connect with cloud-based services and mobile apps [1][2]. This connectivity is very convenient, automated, and accessible, but also increases security threats with the number of connected points that can be attacked [3][4]. Smart home systems can be targeted by attacks like unauthorized access, information theft, device manipulation, and network reconnaissance, which compromise their confidentiality, integrity, and availability. Fig. 1 illustrates

how connected IoT devices, communication technologies, automation, and remote services provide convenience and accessibility while creating a larger networked environment that requires effective security monitoring.

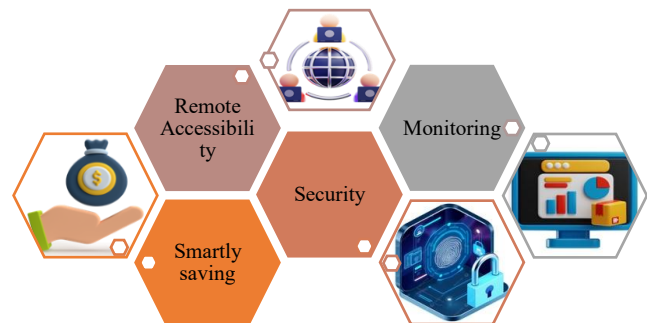


Fig. 1. Key benefits of smart home system.

A smart home can be viewed from a security perspective as a continual process that involves devices creating and sharing data, network traffic gathering and tracking, suspicious activity analysis, and necessary security measures. In the process, an Intrusion Detection System (IDS) is placed to monitor network traffic and detect whether it is normal traffic or if a possible attack is being made. In the context of smart homes, the IDS becomes even more crucial due to the wide range of capabilities and communication methods among IoT devices. A lot of these devices are also constrained in their computing and storage capacity and connected to the network for extended periods of time. Thus, an effective IDS should be able to detect the attacks efficiently without adding unnecessary complexity [5][6].



Fig. 2. Functional benefits of intrusion detection system in smart-home environment.

Fig. 2 summarizes the role of IDS in monitoring network traffic, identifying suspicious activities, distinguishing legitimate and malicious behavior, and supporting timely security response.

Traditional IDS methods rely on signature- and rule-based detection, which are able to detect attacks that have a known pattern. However, they may not be as effective when facing new or changed attack behaviors. Machine learning (ML) methods have therefore received a lot of attention in terms of IoT intrusion detection, as they can learn patterns from network traffic and then identify whether the traffic is legitimate or malicious, based on the patterns learned. In the past, various IDS configurations have been studied, such as centralized, distributed, edge, and fog systems, as well as various machine learning and deep learning algorithms [7][8].

Even with these advances, selecting an appropriate classifier is just part of the intrusion detection problem. The effectiveness and efficiency of the resulting detection model are also directly affected by the representation of the input data. Features in smart home traffic datasets may include many similar features, and there may be others that do not help to differentiate between legitimate traffic and attack. Redundant variables or variables that contribute only a marginal amount of information to the learning problem can make the problem unnecessarily complex and may lead to the creation of an unnecessarily high-dimensional model. Removing features due to correlation can also remove variables that carry useful information for attack classification, on the other hand. Another limitation may arise when evaluating feature relevance based on a single criterion, as different approaches may focus on different aspects of the usefulness of a feature. This shows that there is a need for a feature selection method that takes into account feature redundancy and usefulness to the intrusion detection problem [9].

To address the issue, this study Correlation-Aware Hybrid Feature Selection (CAHFS) model for smart home intrusion detection. The proposed method is intended to remove unnecessary and repetitive features from network traffic data while keeping the information needed to distinguish between normal and attack activity. To assess its effectiveness, the selected feature set is tested with four machine learning models, namely LogisticRegression, DecisionTree, RandomForest, and HistGradientBoosting. The comparison is based on standard classification metrics and error detection analysis using an independent test set.

The study does not propose a new classification algorithm; rather, it focuses on improving the input representation through a correlation-aware hybrid feature selection strategy. The proposed CAHFS model jointly considers feature redundancy and predictive relevance by combining correlation analysis, mutual information, and random forest importance. This provides a broader assessment of feature usefulness than relying on a single selection criterion. The resulting compact feature representation is subsequently evaluated using ML classifiers to determine whether dimensionality reduction can be achieved without substantially compromising the intrusion detection system.

To systematically evaluate the effectiveness of the proposed model, this study addresses the following Research Questions (RQ): -

RQ-1: How does the proposed CAHFS feature representation affect the performance of various ML classifiers for smart-home intrusion detection compared with alternative feature selection strategies?

RQ-2: Which ML classifier provides the most effective balance between detection performance and classification errors when using the proposed CAHFS feature representation?

The following are the main contributions of this study:

- A Correlation-Aware Hybrid Feature Selection (CAHFS) framework is proposed for a smart home intrusion detection system by combining Correlation-based redundancy analysis with complementary mutual information and random forest importance.
- The proposed approach reduces the original 22 features to 16 selected features, providing more compact input space while retaining discriminative information for intrusion detection.
- Four ML classifiers, namely (Logistic Regression, Decision Tree, Random Forest, and HistGradientBoosting), are systematically evaluated using multiple feature selection strategies under separate validation and independent test protocols.
- The classifiers are further assessed using classification metrics, confusion matrix, ROC AUC, and precision-recall analysis to examine both overall predictive performance and class-wise detection behavior.

The remainder of the study is divided into four sections. In Section II, the key research gap was identified. In Section III, the overall methodology adopted was discussed. In Section IV, the interpretation of the experimental analysis was defined and discussed. Finally, in Section V, necessary future directions and research conclusions were discussed.

II. LITERATURE REVIEW

Previous studies have shown that machine learning algorithms can enhance intrusion detection in IoT contexts by detecting network traffic patterns when they are normal or abnormal. Random Forest, XGBoost, and LightGBM prove to be good detection performance for ensemble methods. This study provides a comparison of these methods and presents improved efficiency, specifically 99.92% accuracy overall for the proposed LGB-IDS method [10]. Sophisticated attacks on IoT smart homes are not easily detected by traditional firewalls and intrusion detection systems. Machine learning, deep learning, and ensemble methods have recently been developed to boost detection accuracy and decrease the number of false positives, respectively. Combining these techniques in this study, CNN outperforms them on several datasets, obtaining an accuracy of around 99%, and the hybrid ensemble 99.8% [11]. Existing studies on intrusion detection for smart home systems with IoT typically use simulated data sets or small data sets, and do not consider the IoT devices themselves as an attack vector. This study proposes harmony search optimization of an Extra

Trees classifier for the detection of real-time traffic from 105 devices.

The proposed IDS obtained an accuracy of 99.87% in binary classification [12]. The emerging trend of smart home security systems based on the Internet of Things (IoT) demands efficient facial recognition models for resource-constrained devices. In the era of Internet of Things (IoT) enabled smart home security, efficient facial recognition models are required for resource-constrained devices. This study uses a network architecture named EfficientNet-B4 along with transfer learning and fine-tuning to enhance recognition with a small dataset. The model successfully reached an accuracy of 97% on collected images, whereas the EfficientNet-B4 model had a top-1 and top-5 accuracy of 82.9% and 96.4% [13]. Smart home devices of the IoT generation are becoming more susceptible to DDoS attacks, and it is essential to detect and mitigate such attacks in real time. This study suggests an SDN-based architecture that integrates the machine learning scheme for traffic classification on the IoT devices with the SNORT traffic analysis scheme for controller protection. SVM, Logistic Regression, Decision Tree and KNN were tested and among them Decision Tree has the maximum accuracy of 99% [14]. With IoT and machine learning, home security becomes intelligent, with the ability to detect threats in real time based on the information from cameras and movement sensors. Three models, namely Decision Tree, transfer learning, Logistic Regression, and SVM are evaluated in this study. The system proposed in this work was successful in reaching an accuracy of 99.2%, while Decision Tree and transfer learning performed better than SVM and Logistic Regression [15]. Anomaly detection is crucial in human activity for smart homes, especially when residents are elderly, and they may not be monitored constantly. The environment of multi-resident facilities adds to the complexities, and comprehensive anomaly datasets are limited. In this study, eight machine learning and two deep learning classifiers are compared, and it is concluded that GRU is the best when it comes to distinguishing abnormal and normal activities [16]. Smart home technologies have improved the convenience of the home, but opened up a great deal of security concerns, which are to be tackled by a good intruder detection system. The signature-based, anomaly-based, and hybrid IDS methods are the existing ones. Their interoperability with machine learning, IoT, and edge and cloud computing will be explored with a focus on dynamic, privacy-sensitive security solutions [17].

Research Gap: An overview of these studies reveals that most of the current studies are dedicated to the selection of an effective classifier and to the assessment of its detection performance, whereas relatively little work has been done on reducing the number of features without loss of useful information. The studies also process the information of features in various ways. In [10, 14, and 15] the different classification algorithms are explored mainly, while, in [11] the deep learning and ensemble methods are combined, and in [12] an optimization technique is applied in conjunction with an Extra Trees classifier. All of these studies look at the potential of various learning methods, but none of them explore feature redundancy and feature relevance in combination in a single

feature-selection method. If removal of highly correlated features is undertaken alone, then it does not indicate which features, if any, from the remaining features have the most influence on target class prediction. This gap inspires the proposed CAHFS approach, which combines these two aspects: correlation analysis is applied to solve the redundancy problem, and Mutual Information and Random Forest feature importance are used to find informative features.

To overcome these problems, this study proposes a hybrid feature-selection method entitled CAHFS, which is based on the combination of correlation analysis, Mutual Information, and the feature importance of Random Forest. The goal is to eliminate features that are not needed while preserving features of the smart-home network traffic that are useful for intrusion detection. The resulting feature set is then evaluated on several machine-learning classifiers to assess the overall predictive accuracy of the classifiers and the error patterns.

III. METHODOLOGY

In this section, the methodology that has been adopted towards the development of the proposed smart home intrusion detection framework is presented. Data collection and data preprocessing are the first steps in the process, followed by exploratory data analysis to gain insights into the nature of the network traffic. The main methodological part is then a correlation-aware hybrid feature selection (CAHFS) that selects the most useful and least redundant features. Selected features are then applied to build and test various machine learning models under the same experimental conditions to compare them. As a general principle, the proposed workflow is designed to minimize the original feature space whilst preserving the information required for reliable intrusion detection and ensuring a fair comparison between the models evaluated. Fig. 3 illustrates the steps adopted.

A. Data Collection

The dataset used in this study was retrieved from a publicly available source [18] The dataset was captured under normal operation as well as from 7 different multi-stage attack scenarios that can arise in a smart home environment that hosts end devices of heterogeneous nature. The dataset was published in 14 January 2026 by Amrita Vishwa Vidyapeetham. The dataset comprised (178831 x 22).

The distribution of training and validation data for the two classes in the dataset is shown in Fig. 4. The test set contains 29,872 samples and was kept separate from the model development process for final evaluation. Approximate Class ratio: 57348 normal samples, 91611 attack samples (1.591). The data set is moderately imbalanced as the attack samples outnumber the normal samples. To minimize the impact of this imbalance during the training of these models, SMOTE (Synthetic Minority Over-sampling Technique) was used to augment the training data. This would help to ensure that the minority class was represented and the likelihood of the models becoming biased towards the majority class was minimized.

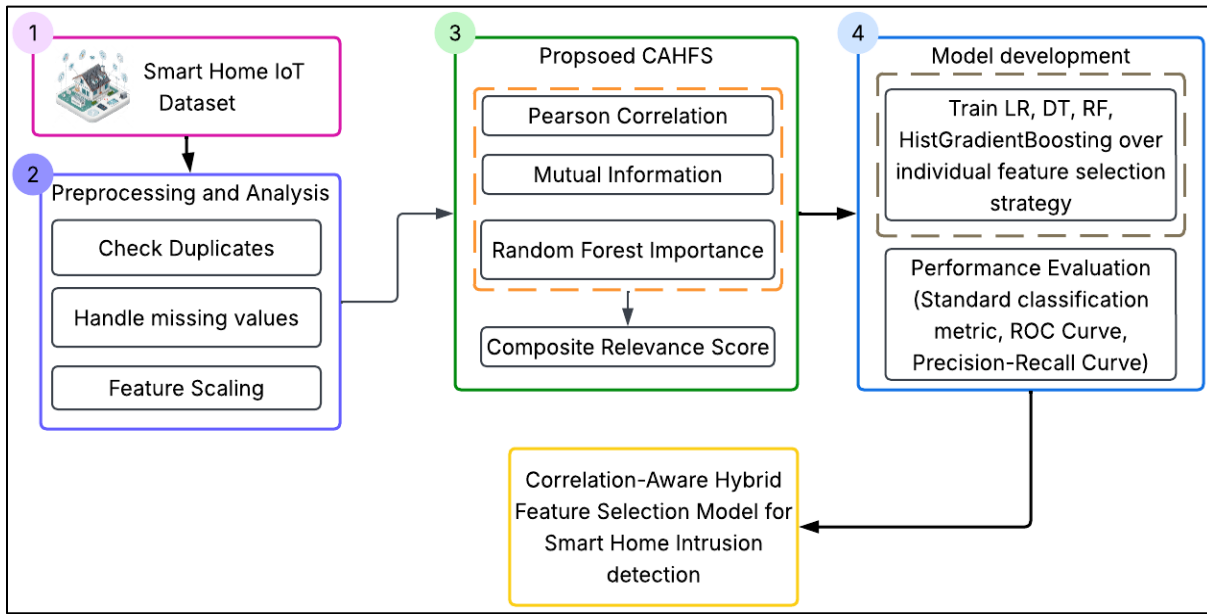


Fig. 3. Overall research pipeline.

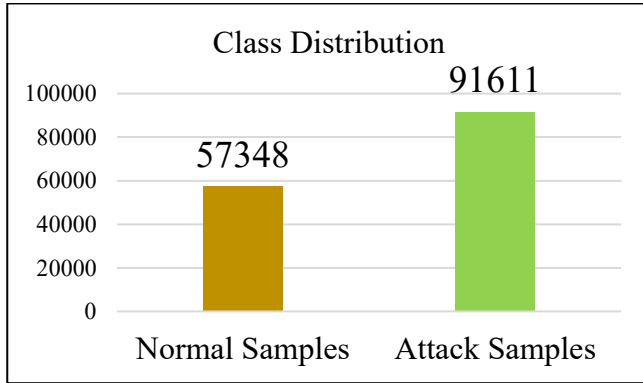


Fig. 4. Class distribution of training and validation data.

B. Preprocessing and Analysis

The raw network traffic is first cleaned and formatted to be consistent before creating the classification models. Records are duplicated and deleted as needed. All infinite values are replaced with missing values, and all values missing after that are processed in a preprocessing procedure that is tailored only to the training data. The training portion is then split into separate training and validation data sets with stratified sampling. This means that the number of normal instances is the same in both sets as the number of attack instances [19][20].

C. Proposed Correlation-Aware Hybrid Feature Selection (CAHFS)

In the context of smart home intrusion detection, feature selection plays a crucial role as network traffic datasets can frequently be large and include numerous flow and statistical features, some of which may lack discriminatory power or be redundant. This work presents a novel approach named Correlation-Aware Hybrid Feature Selection (CAHFS) to minimize the above redundancy and enhance the learning efficiency. CAHFS rates features based on three criteria: relevance to the intrusion class, importance in predicting the

class, and redundancy with the other features selected. Mutual Information (MI) is a measure of target relevance, and Random Forest (RF) importance is a measure of predictive contribution. Pearson correlation is then applied to determine whether there is information that is shared between features. These criteria are aggregated into a score, with informative features getting a higher score and highly redundant features getting a lower score, with the criteria being correlation-aware. The procedure consists of four steps: 1) correlation analysis, 2) MI-based relevance estimation, 3) RF-based importance estimation, and 4) correlation-aware score fusion. This ranking is then applied to determine a small number of useful features to be used for the intrusion detection models.

Let the traffic in the dataset $\mathcal{D}$ be denoted by X and the available features be represented as d the corresponding intrusion classes be shown as Y_i . Hence, the original feature space and training dataset is given by Eq. (1) and (2), respectively:

$$\mathcal{F} = \{X_1, X_2, \dots, X_d\} \quad (1)$$

$$\mathcal{D}_{tr} = \{(X_i, Y_i)\}_{i=1}^N \quad (2)$$

1) *Pearson correlation analysis*: In this stage, the proposed CAHFS method examines the relationship between every pair of traffic features using Pearson correlation [21]. The correlation between features X_i and X_k is calculated by:

$$r_{jk} = \frac{\sum_{i=1}^N (x_{ij} - \bar{x}_i)(x_{ik} - \bar{x}_k)}{\sqrt{\sum_{i=1}^N (x_{ij} - \bar{x}_i)^2 \sum_{i=1}^N (x_{ik} - \bar{x}_k)^2}} \quad (3)$$

A feature pair is highly correlated when $|r_{jk}| \geq 0.90$, thus the highly correlated feature set is defined as:

$$\mathcal{C} = \{(i, k): |r_{jk}| \geq 0.90\} \quad (4)$$

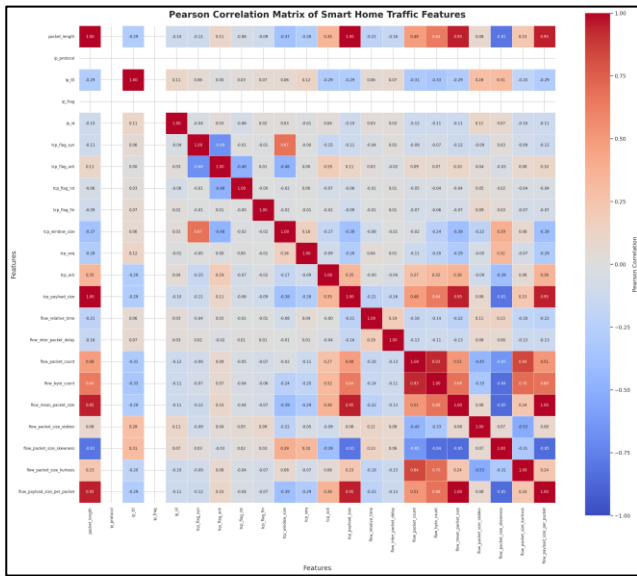


Fig. 5. Selected features using Pearson correlation method.

The training data was used to calculate the Pearson correlation matrix as illustrated in Fig. 5. It gives an idea of the linear relationships between the traffic features themselves. Correlation values near +1 or -1 signify a strong correlation, and values near 0 signify that there is little or no linear correlation. For this analysis, feature pairs having an absolute correlation coefficient value higher than 0.90 were considered highly correlated and redundancy was noted during feature selection.

This step identifies the potential redundancy among smart home traffic features.

2) *Mutual Information (MI)*: After identifying the correlated features, mutual information is calculated to determine how strongly each feature is associated with intrusion class [22]. For feature X_j and Y represents the intrusion class, the MI score is calculated as:

$$I_j = I(X_j; Y) \quad (5)$$

A larger I_j indicates that the features provides more information about the intrusion class. The normalization of the MI score is defined as:

$$\bar{I}_j = \frac{I_j - I_{min}}{I_{max} - I_{min}} \quad (6)$$

The scores of the input features are shown in terms of the Mutual Information with the intrusion class in Fig. 6. The higher the MI, the more information a feature contains about the class it represents and therefore the more value it is for discriminating between normal and attack traffic. Unlike Pearson correlation, which is primarily used in detecting linear relationships, Mutual Information can also detect nonlinear dependencies. This can be helpful in recognizing informative features that might not have been captured by the correlation analysis.

This step is important because MI can capture nonlinear relationships that may not be captured through the Pearson correlation method.

3) *Random forest feature importance*: A balanced Random Forest classifier is trained using the training dataset. The predictive contribution of each feature is obtained from the trained ensemble [22][23]. For feature X_j , its random forest importance is defined by:

$$\mathcal{F}_j = RFimportance(X_j) \quad (7)$$

The normalization score of $\mathcal{F}_j$ is expressed by:

$$\bar{\mathcal{F}}_j = \frac{\mathcal{F}_j - \mathcal{F}_{min}}{\mathcal{F}_{max} - \mathcal{F}_{min}} \quad (8)$$

A larger $\mathcal{F}_j$ indicates that feature contributes more strongly to the predictive decisions for Random Forest.

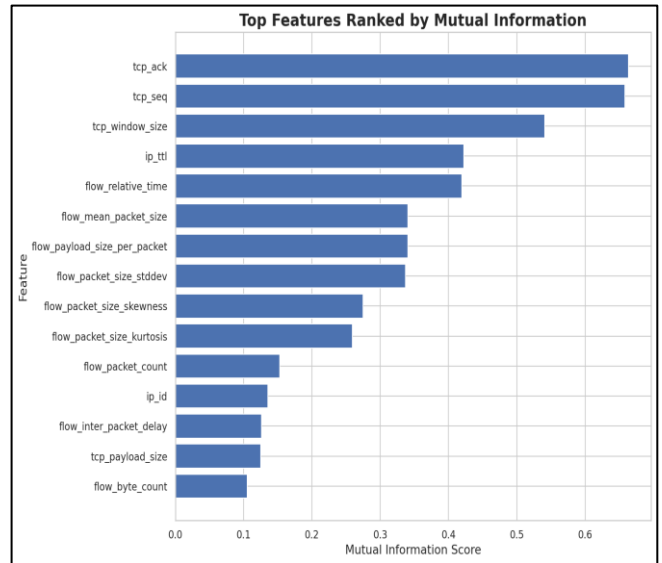


Fig. 6. Top features ranked by MI method.

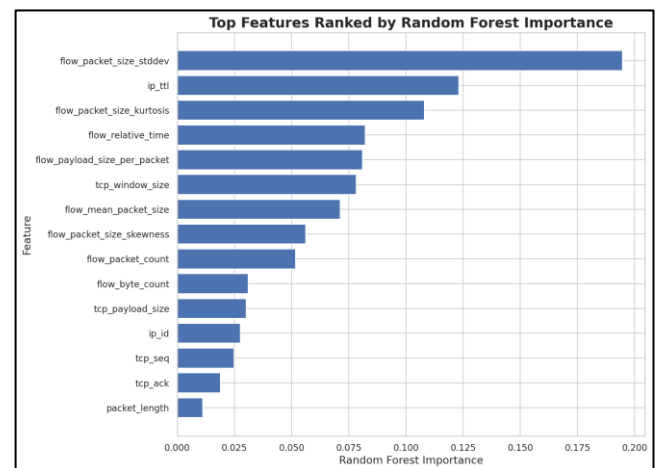


Fig. 7. Random forest importance features.

The importance scores obtained by the Random Forest model using the training data is shown in Fig. 7. The larger the importance score, the more the feature influences the prediction of the model; the smaller the score, the less the feature influences the model. This ranking is another perspective on feature usefulness in addition to Mutual Information and correlation

analysis, since Random Forest can discover nonlinear patterns in the data

4) *Proposed CAHFS-based composite score*: For feature $\mathcal{S}$ represent the set of features that have already been selected. For a candidate feature X_j its correlation redundancy with the selected features is defined as:

$$\mathcal{R}_j = \max_k \in \mathcal{S} [r_{jk}] \quad (9)$$

Finally, the normalized MI and RF scores are then combined using equal weights to obtain the final composite score for each feature:

$$\mathcal{S}_j^{CAHFS} = \frac{1}{2} \bar{I}_j + \frac{1}{2} \bar{\mathcal{F}}_j \quad (10)$$

where, $\bar{I}_j$ represents the normalized MI score, while $\bar{\mathcal{F}}_j$ denotes the normalized Random Forest Importance score. $\mathcal{S}_j^{CAHFS}$ shows the final CAHFS composite score of features X_j .

This means that a feature is preferred when it is informative about the intrusion class, useful for prediction, and not excessively redundant with features that have already been selected.

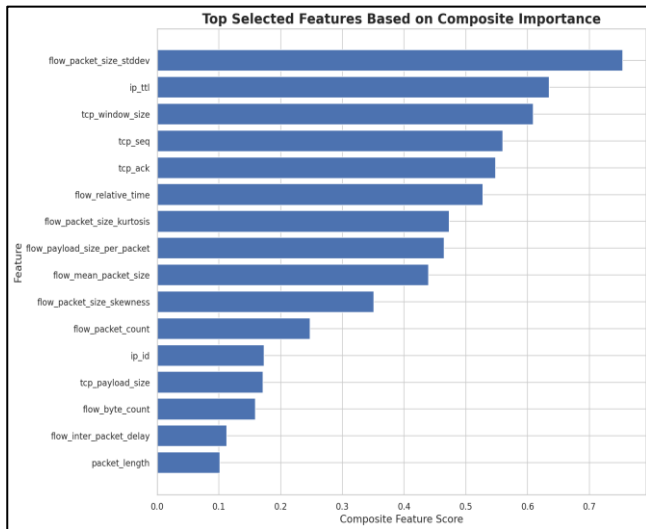


Fig. 8. Composite importance of selected features.

The final feature ranking derived by CAHFS (after merging the relevance scores obtained from Mutual Information with feature importance scores from Random Forest) is presented in Fig. 8. The higher the combined scores for each feature, the closer it will be at the top of the ranking, and the closer it is to the top, the more likely it is to be selected according to the predetermined retention criteria. The reduced feature set is then fed into all the classification models, and the experimental conditions are kept the same during the evaluation.

The final number of features was rounded to the nearest whole number and 16 features remained out of an initial set of 22 features. For consistency, the same retention threshold was used in all the feature-selection experiments. In this study, there is no systematic testing of different retention levels, and this is identified as a limitation and is suggested for further work to test

the effects that various sizes of feature sets have on the classification performance-dimensionality tradeoff.

Algorithm-1: Proposed CAHFS Method

Input: training feature matrix $X \in \mathbb{R}^{N \times d}$, training labels $Y \in \{0,1\}^N$, correlation threshold $\tau = 0.90$, Feature retention ratio $r = 0.70$

Output: selected Feature Subset $\mathcal{S}$

Start:

1. Load $X \in \mathbb{R}^{N \times d}$ and $Y \in \{0,1\}^N$
 2. Compute Pearson correlation Matrix $\mathcal{R}_j$ using Eq. (3)
 3. Identify highly correlated feature pairs using Eq. (4)
 4. Compute Mutual Information between each feature and target labels using Eq. (5)
 5. Train a Random Forest classifier using X and Y
 6. Extract Feature Importance using Eq. (7)
 7. Normalize the MI and RF scores using Min-Max normalization
 8. Calculate the final composite CAHFS score using equal weights 0.50 for MI and RF, as defined in Eq. (10)
 9. Rank all features in descending order according to their CAHFS score.
 10. Determine the number of retained features using the predefined retention ratio:
 11. $N_{SELECTED} = \lceil r N_{features} \rceil$
 12. Select Top-ranked features
 13. Return $\mathcal{S}$
- End
-

The proposed Correlation-Aware Hybrid Feature Selection (CAHFS) procedure is described in Algorithm 1. It first calculates the Pearson correlation matrix for the training data and finds the pairs of features that are highly linearly correlated. The relevance and importance of each feature with respect to intrusion detection is then measured separately using Mutual Information and Random Forest. The scores from both measures have been normalized; the scores of the two measures are then pooled together with equal weights to get the overall relevance score. This combined score is used to rank the features and the top 70% are selected as the final set of features. These selected features are then given to the machine-learning models, which would then be trained and evaluated in turn.

D. Model Development

Four machine-learning models are used to evaluate the selected 16 features. The models are linear classification, single tree learning, bagging, and gradient boosting.

1) *Logistic regression*: As a linear baseline to investigate whether the selected features can distinguish between normal and attack traffic with a simple decision boundary [24].

2) *Decision tree*: Recursively partitions the feature space based on decision rules, thus enabling the model to learn nonlinear attack patterns [25].

3) *Random forest*: Aggregates several decision trees to achieve greater stability and to pick up more interactions among the features [26].

4) *HistGradientBoosting*: An efficient model, based on the gradient boosting framework, to learn non-linear relationships

using the histogram technique and to yield an excellent model with the selected feature set [27].

TABLE I. HYPERPARAMETERS

Models	Parameters	Description
LR	C	0.1
	Penalty	L2
RF	n_estimators	100
	Max_depth	None
	Min_samples_split	2
	Min_samples_leaf	1
DT	Max_depth	10
	Min_samples_split	2
	Min_samples_leaf	1
	Entropy	
HistGradientBoosting	Learning Rate	0.1
	Max_iteration	150
	Max_leaf_nodes	31
	l2_regularization	0.1

The parameter values used for the four machine-learning classifiers in the experiments are shown in Table I. These settings were kept constant for all feature-selection tests, enabling the comparison of the different models under the same conditions. The validation set was used during the development of the model for evaluation and tuning, and the test set was not used until the final evaluation. These details have been added to the revised manuscript to adequately describe the experimental setup and to aid in the reproduction of the study. Logistic Regression: L2 regularization (C=0.1). Random Forest employed 100 trees and did not specify the maximum depth of the trees. The maximum depth was 10 for the Decision Tree, and entropy was chosen as the splitting criterion. The model, named HistGradientBoosting, was trained with a learning rate of 0.1, 150 boosting iterations, 31 maximum leaf nodes, and L2 regularization of 0.1. The same settings were used in all experiments to ensure as far as possible, a consistent comparison among the various feature subsets.

IV. RESULTS AND DISCUSSION

Experiments were all performed in Python, and all the usual machine-learning libraries were used for data pre-processing, feature selection, model development, and evaluation of results [28]. The data set was split into training, validation, and test sets

TABLE IV. CLASS-WISE ERROR CHARACTERISTICS AND SPECIFICITY OF THE EVALUATED ML CLASSIFIER ON UNSEEN TEST SET

Model	True Negative	False Negative	True Positive	False Positive	Specificity	Overall error rate
LR	12160	6116	6574	5022	0.707	0.372
DT	15333	2451	10239	1849	0.892	0.143
RF	15437	4218	8472	1745	0.898	0.199
HistGradientBoosting	16337	1834	10856	845	0.950	0.089

Table IV summarizes the errors of the evaluated classifiers on the test set that had not been included in the development of

to maintain consistency and an unbiased experimental procedure. All the parameters of the preprocessing were set by the training data alone, and SMOTE was used to solve the class imbalance problem in the training set. Feature selection was also performed based on the training data only. Models were developed using the training set, and the validation set was used to select and tune the model parameters. The test set was held entirely apart from the entire development process and used only for the final evaluation. The data used was made up of 119,167 training samples, 29,792 validation samples, and 29,872 test samples as seen in Table II. By not changing the test set during model development, the final results were able to give a more accurate indication of the degree of generalization of the models evaluated.

TABLE II. DISTRIBUTION OF SAMPLES

Dataset Partition	Samples	Percentage
Training	(119167, 16)	66.64%
Validation	(29792, 16)	16.70%
Testing	(29872, 16)	16.66%

TABLE III. PERFORMANCE OF ML CLASSIFIERS USING PROPOSED CAHFS

ML models	Accuracy	Precision	Recall	F1 score
LR	62.71%	56.69%	51.80%	54.14%
RF	80.04%	82.29%	66.76%	74.97%
DT	85.61%	84.70%	80.69%	82.65%
HistGradientBoosting	91.03%	92.78%	85.55%	89.02%

Table III represents the performance of ML classifiers using the proposed CAHFS. Among classifiers HistGradientBoosting classifier outperforms across all metrics as compared to other models by achieving an accuracy (91.03%), precision (92.78%), recall (85.55%), and f1 score (89.02%). Its superior performance can be attributed to its nonlinear feature interactions and complex decision boundaries, which are important for distinguishing normal and attack traffic. Followed by DT, which exhibits accuracy (85.61%), precision (84.70%), recall (80.69%), and F1 score (82.65%). The RF classifier performs average, obtaining an accuracy (80.04%), precision (82.29%), recall (66.76%), and F1 score (74.97%). The LR classifier performed poor by achieving the lowest accuracy (62.71%), precision (56.69%), recall (51.80%), and F1 score (54.14%), likely because its linear decision boundary is less effective for the complex relationship present in the selected features.

the classifiers. HistGradientBoosting had the best performance of all four models. It had the lowest false-positive rate (4.92%)

and the lowest false-negative rate (14.45%), the highest specificity (95.08%), and the lowest overall error rate (8.97%). This in turn allowed the model to successfully prevent false alarms and missed attacks on normal traffic. DT gave the next best results of a 19.31% false-negative rate, 10.76% false-positive rate, and 14.39% error rate. The error rate for Random Forest was higher (19.96%) due to a higher false-negative rate of 33.24%, even though it had a slightly lower false-positive rate (10.16%). Logistic Regression did the worst with false-positive and false-negative rates of 29.23% and 48.20%, respectively. In general, the results obtained indicate that HistGradientBoosting gave the best error profile, especially with the number of false alarms and the number of missed intrusion cases.

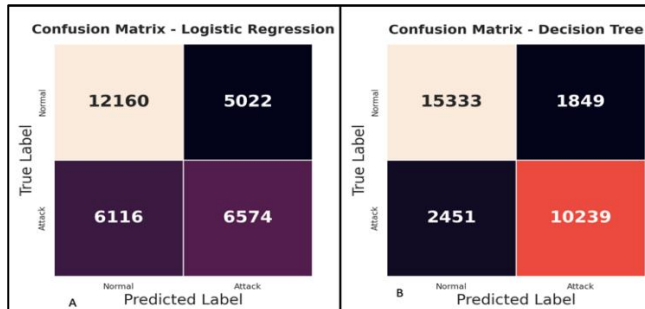


Fig. 9. Confusion matrix for LR and DT.

Fig. 9 illustrates the performance of LR and DT models over unseen test samples. In Fig. 9 (A), the confusion matrix for LR is shown, where the model exhibits lower performance, correctly classifying (18734) samples out of (29872), while (11138) samples were misclassified. Conversely, in Fig. 9(B), the performance of the DT classifier is shown. From the total samples (29872), 25572 samples were correctly predicted; only 4300 samples were misclassified.

Fig. 10 represents the performance of RF and HistGradientBoosting classifiers over test samples. In Fig.

10(A) the performance of RF model is depicted where the model correctly classified (23909) samples from total (29872) while 5963 samples were wrongly classified. On the other hand, Fig. 10(B) shows the confusion matrix of the HistGradientBoosting classifier, where the classifier remarkably classified (27193) samples with only 2679 misclassifications.

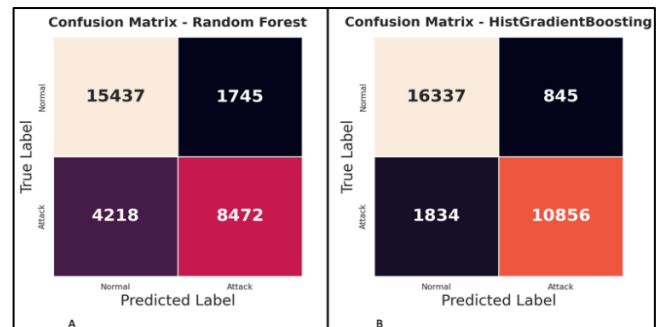


Fig. 10. Confusion matrix for RF and HistGradientBoosting.

The ROC curves for the four ML classifiers using the sets of features selected by the proposed CAHFS method are presented in Fig. 11. HistGradientBoosting (0.964) performed best with a ROC-AUC value, followed by RF (0.910) and DT (0.848). LR had the lowest ROC-AUC of 0.669, which clearly displayed its separation capability between two classes. The results show that HistGradientBoosting performed better than the other models for distinguishing between the desired classes for a variety of classification thresholds. It may be because its higher ROC-AUC allows it to learn the nonlinear relationship and interaction among the selected features. The trees are constructed on a sequential basis, with each new tree correcting the errors made by previous trees, which can enhance the overall classification performance. The opposite of this is true for the lower ROC-AUC value achieved by LR, which indicates that there is not a good linear association between the features used and the class labels.

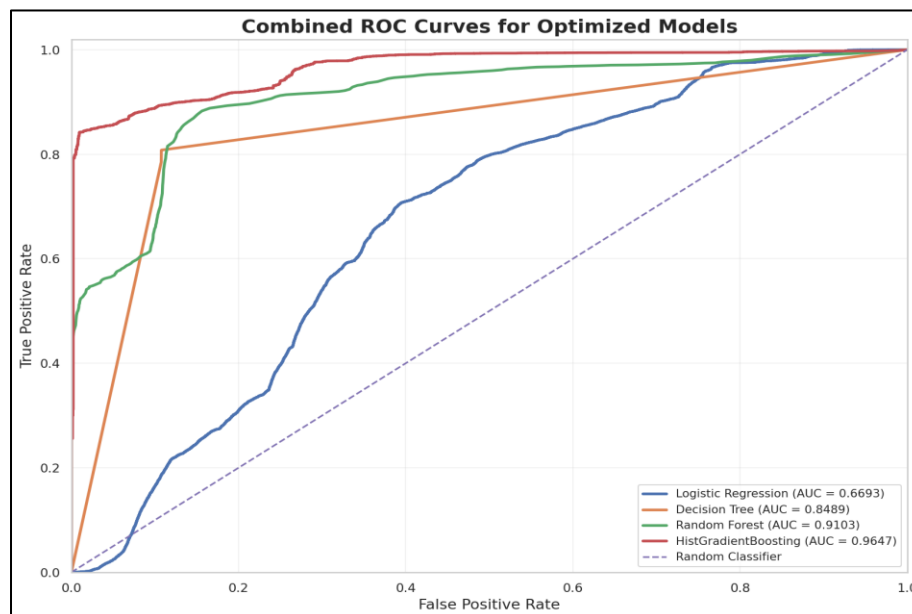


Fig. 11. ROC Curve of ML classifier using proposed CAHFS.

TABLE V. COMPARATIVE EVALUATION OF ML CLASSIFIERS USING ALTERNATIVE AND PROPOSED FEATURE-SELECTED STRATEGIES

Feature Methods	No Features	Models	Accuracy	Precision	Recall	F1 score	ROC-AUC
All	22	LR	63%	60%	61%	60%	0.673
		DT	86%	84%	83%	83%	0.855
		RF	80%	82%	67%	74%	0.916
		HistGrdaientBoosting	89%	90%	85%	87%	0.955
Pearson	18	LR	63%	59%	49%	53%	0.647
		DT	86%	86%	82%	84%	0.863
		RF	82%	87%	68%	76%	0.933
		HistGrdaientBoosting	92%	95%	85%	90%	0.9605
MI Only	16	LR	62%	56%	51%	54%	0.669
		DT	86%	85%	81%	83%	0.855
		RF	80%	83%	66%	74%	0.914
		HistGrdaientBoosting	90%	92%	85%	88%	0.964
RF Only	16	LR	62%	56%	51%	54%	0.669
		DT	86%	85%	81%	83%	0.855
		RF	80%	83%	66%	74%	0.914
		HistGrdaientBoosting	89%	89%	85%	87%	0.963
Proposed CAHFS	16	LR	62%	60%	59%	59%	0.66
		DT	80%	82%	79%	80%	0.85
		RF	85%	84%	80%	81%	0.91
		HistGrdaientBoosting	91%	91%	90%	90%	0.96

The performance of the four machine-learning classifiers in the three feature selection approaches (conventional feature selection, the proposed CAHFS approach, and the full set of features) is compared in Table V. Overall, the best results are obtained with HistGradientBoosting for varying feature configurations. It was able to reach an accuracy of 89%, an F1-score of 87%, and a ROC-AUC of 0.955 when using all 22 features. After feature selection using Pearson, accuracy and F1 score were 92% and 90%, respectively, while the number of selected variables decreased to 18. Further, the number of features was reduced to 16 variables by Mutual Information or RF selection, which yielded similar high ROC-AUC values of 0.964 and 0.963, respectively, in HistGradientBoosting. The proposed CAHFS method also generated competitive outcomes using just 16 features. With the HistGradientBoosting model, it achieved 91% accuracy, 90% recall, 90% F1 score, and a ROC-AUC of 0.960. The results indicate that the CAHFS can eliminate less useful variables without compromising the important information for classification. That is, the feature set is reduced, but information is still retained and input representation is compact. The superior results of HistGradientBoosting over LR, DT, and RF suggest that there are possible nonlinear relationships or interactions between the features in the data. A gradient-boosting approach is better suited than a linear classifier or traditional tree-based models to detect such patterns.

CAHFS provides slightly less accurate and precise classification, but with two fewer features. This indicates that the main advantage of CAHFS lies in its ability to provide a more compact and efficient feature representation rather than in

achieving the highest accuracy. The overall results indicate that the reduction of input features while preserving the information required for reliable classification is possible with CAHFS.

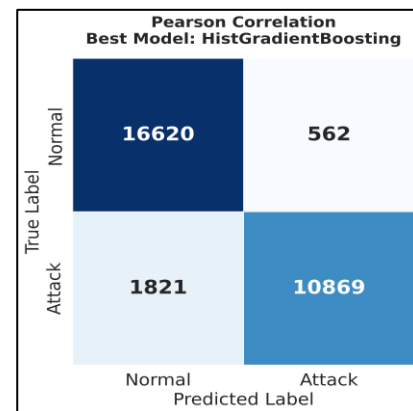


Fig. 12. Confusion matrix for the best-performing model.

The confusion matrix for the best-performing experimental setup (Pearson Selected-HistGradientBoosting) is presented in Fig. 12. The matrix shows that the model was able to correctly predict the majority of the normal and attack samples, with a relatively small number of false predictions. This is in line with the very good accuracy and F1-score of the model. The results also indicate that some redundant information was removed by using the features selected by the Pearson method without the loss of the features important for distinguishing between the two classes. This led to HistGradientBoosting having a good classification result on fewer input variables.

The ROC curve of the Pearson-selected HistGradientBoosting configuration is presented in Fig. 13. The curve remains in the upper left part of the plot, suggesting that the model has the ability to distinguish between normal and attack samples for various decision thresholds. The corresponding ROC-AUC value is also a sign of its overall discriminatory power. The ROC-AUC scores are higher for classifiers that can achieve a good balance between true positives and false positives for a range of thresholds. This indicates that the feature set chosen has the information necessary for the effective detection of intrusions.

The precision-recall curve of the Pearson Selected-HistGradientBoosting is shown in Fig. 14. The curve depicts the changes in precision and recall with varying classification thresholds. The analysis is particularly relevant in this context to investigate the ability of the model to find attack samples and keep the rate of incorrect positive predictions under control. The accuracy of the classifier's performance in the precision-recall graph shows that it can identify a significant percentage of attack samples with only a modest number of false alarms. In addition to the ROC analysis, this gives a better sense of the model's positive class identification ability.

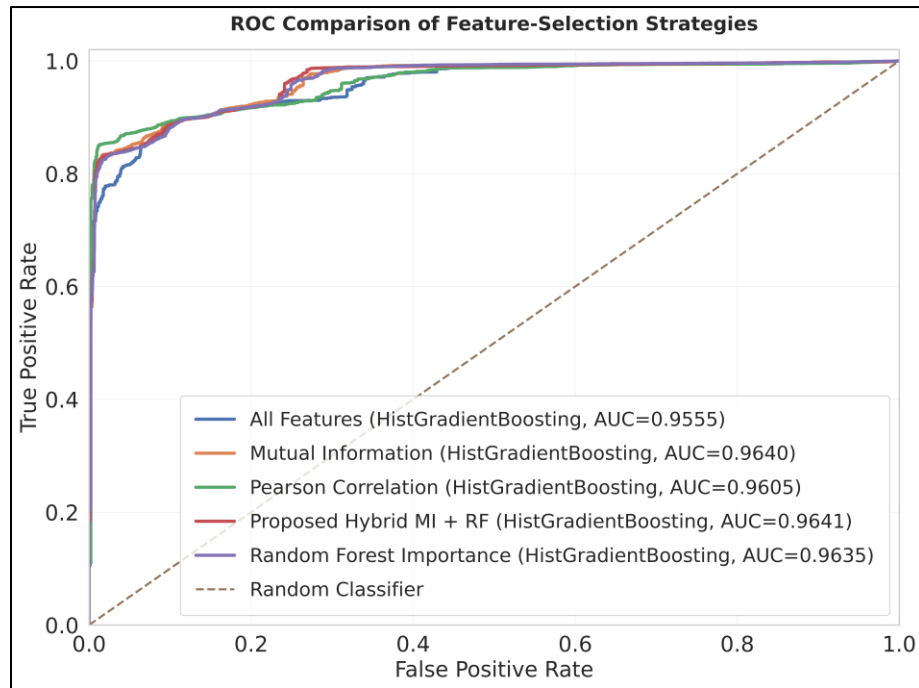


Fig. 13. ROC curve analysis of the best-performing model.

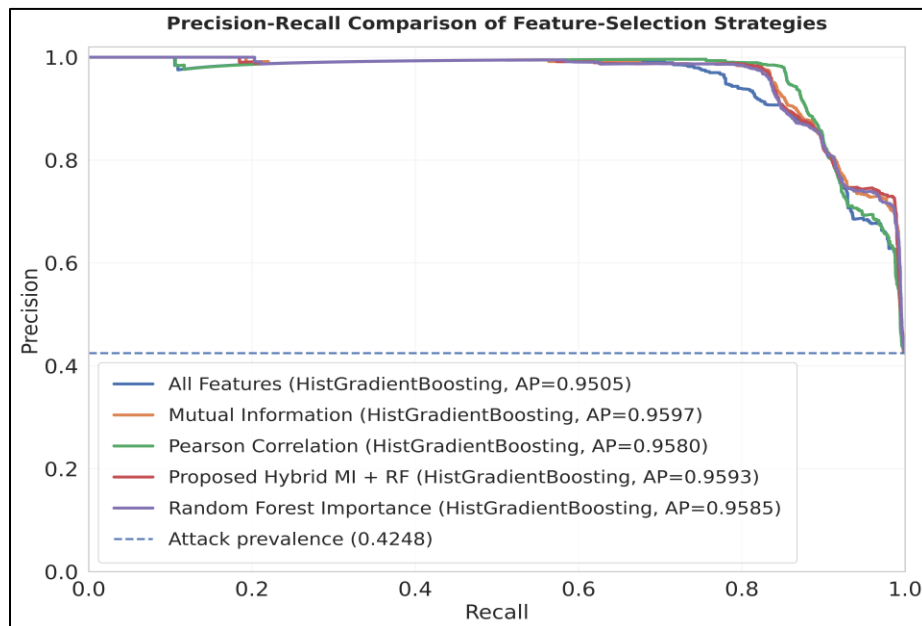


Fig. 14. Precision-recall curve of the best-performing model.

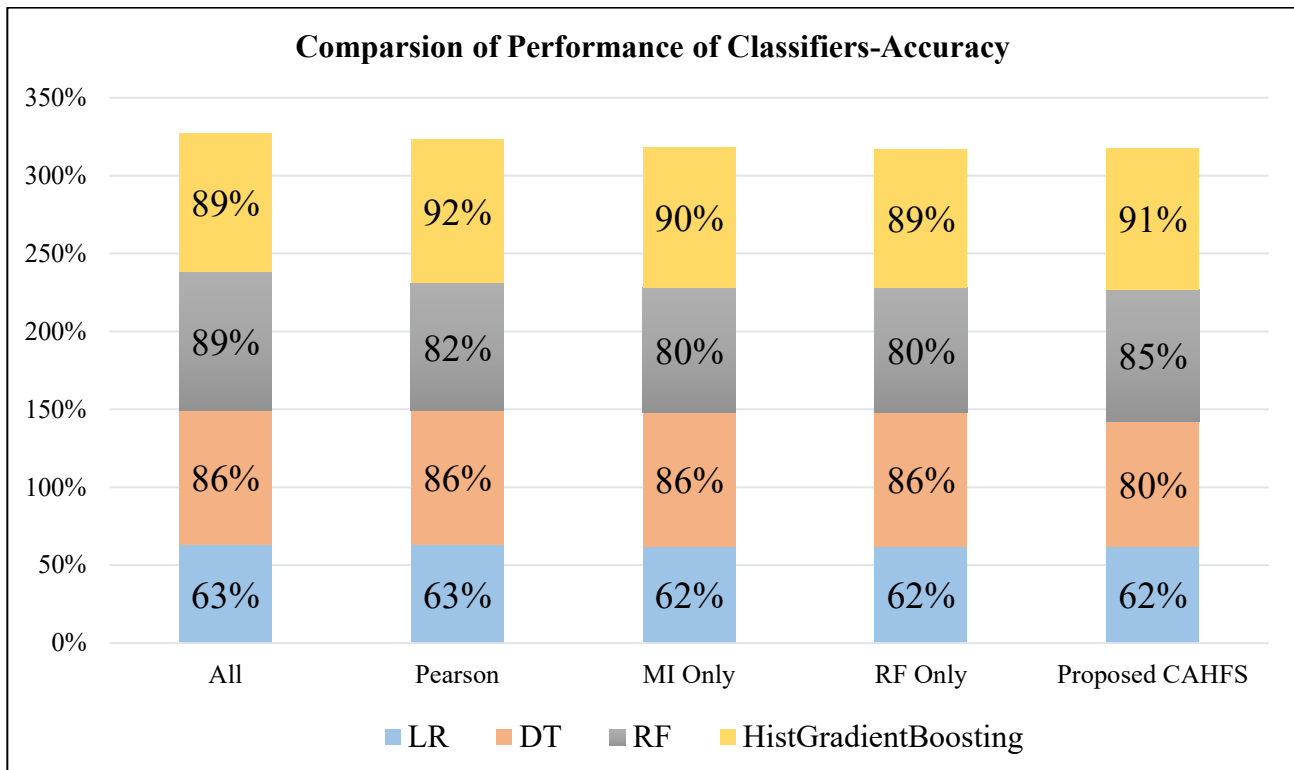


Fig. 15. Comparison of model performance across accuracy.

The results of the machine-learning models evaluated are compared in Fig. 15 under the various feature-selection strategies. Results indicate that the effect of feature selection was different for the classifiers. The Logistic Regression model performed relatively poorly over the tested feature sets while HistGradientBoosting performed the best. It is also observed from the comparison that a decrease in the number of features did not necessarily mean that the classification results are worse. In a few instances, the elimination of less informative or redundant variables resulted in better model performance. The proposed CAHFS method also yielded competitive results, with only 16 features, demonstrating that the selection of a set of features can provide a compact representation without loss of information relevant to intrusion detection.

V. DISCUSSION

The experimental results demonstrate that the proposed correlation-aware hybrid feature selection (CAHFS) framework is able to reduce the input feature number without significant degradation in ID performance. Overall, HistGradientBoosting performed best when compared to the other classifiers, with an accuracy, precision, recall and F1-score of 91%, 91%, 90%, and 90%, respectively, as reported in Table III. The values indicate that the model could successfully detect the attacks and had a low rate of false alarms. Logistic Regression, however, did not do as well – indicating that the patterns in the logistic regression model between the features of the network traffic and the attack classes are not well described using only a linear decision boundary.

The difference is also evident in the results of the tests reported in Table IV. HistGradientBoosting had the best overall

error rate of 8.9% and the highest specificity of 95.0%. It also made fewer misclassifications than the other classifiers. This is crucial for a smart home intrusion detection system, as false alarms can make a security system hard to manage, and missed attacks can let harmful activity slip through the cracks. These results are confirmed by the confusion matrices displayed in Fig. 9 and 10, which illustrate the accuracy of the models at predicting the test samples for the two types of classification outlined above.

Regarding RQ1, the results indicate that CAHFS was able to classify the same number of features as the original with a modest drop in classification accuracy. The accuracy was slightly higher with Pearson-based selection, but CAHFS still obtained a ROC-AUC of 0.96 using fewer features. This indicates that the proposed hybrid technique was able to remove less useful information and retain the features that are important for distinguishing between normal and attack traffic.

This can be observed in the ROC analysis shown in Fig. 11. For the ROC-AUC scores, HistGradientBoosting was able to distinguish normal from attack traffic for most classification thresholds, with an AUC score of approximately 0.96. A possible explanation for this better performance is that it can derive non-linear relationships between features. The interactions among multiple variables may be present in the network traffic produced by a smart-home environment, and may not be well captured by a linear model.

From results presented in Table V, the selection of the feature-selection method also had a significant impact on the results of the classification. The best accuracy, 92%, was obtained by the Pearson-based selection. Despite this, the

original 22-feature representation was reduced to just 16 features, yielding competitive results of 91% accuracy, 90% recall, 90% F1 score, and a ROC-AUC of 0.96. Fig. 12- 15 show the individual results of the model performance, including, among others, the confusion matrix, the ROC behavior, the precision-recall relationship, and a comparison of the various feature-selection strategies.

For RQ2, the overall best performance was achieved by the HistGradientBoosting classifier. This ROC-AUC, specificity (around 95%), and overall error rate (around 8.9%) suggest that it performed better at separating the attack traffic from the normal traffic and with a relatively low rate of misclassifications. The results indicate that HistGradientBoosting performed better for the feature representation in this study than LR, DT, and RF.

It should be noted that the results shown in Tables III-V were conducted with a single training, validation, and testing split. It is important to note that the relatively small differences between the feature-selection methods, particularly between the one-two percentage-point differences in accuracy, should not be interpreted as a statistically significant advantage. Repeated experiments or cross-validation with variance estimations would be necessary to draw such conclusions. All feature-selection methods and classifiers were, however, tested on the same split to provide a basis for comparison. Therefore, the single split used in this study is recognized as a limitation in this study. Future work will consider multiple splits (or cross-validation) to get a better idea of how they perform.

1) *Practical implication*: This decrease in the number of features is also useful for practical use. A compact feature set can minimize the data processing and storage requirements, a benefit for smart-home gateways and other IoT devices that have restricted computing power. Meanwhile, the increased specificity achieved by the more powerful models could help to reduce false positive alerts for security. The overall detection performance also indicates that the proposed method may be beneficial for the real-time monitoring of network traffic in smart-home scenarios; however, a further evaluation is necessary in realistic deployment scenarios.

2) *Limitations*: Some restrictions remain to be noted. The experiments were done with one dataset, so the results might not reflect the performance of CAHFS in other smart home scenarios or attacks. We further evaluated the feature-selection method using a pre-defined set of hand-crafted machine-learning classifiers in one experimental configuration. Other investigations may be carried out to assess the method with various datasets, to compare the method with a broader set of feature selection methods, and to look into adaptive as well as deep learning-based intrusion detection approaches.

VI. CONCLUSION

This study proposed a novel hybrid feature selection (HFs) scheme, termed as Correlation-Aware Hybrid Feature Selection (CAHFS), for machine learning-based intrusion detection (IDS) in smart-home IoT networks. The primary objective was to minimize the number of features that the classifiers use while retaining as much information as possible for differentiating

between normal and attack traffic. To analyze the impact of feature selection on the classifiers, four machine learning classifiers were tested with various feature sets. Overall, the best-performing model was HistGradientBoosting. It had 16 features in the CAHFS feature set and its accuracy, precision, recall, F1-score, and ROC-AUC were 91%, 91%, 90%, 90%, and 0.96, respectively. The model also achieved around 95% specificity and an overall error rate of 8.9%. These results were supported by the confusion matrix and ROC results, which also demonstrated that the model was able to correctly classify normal and attack traffic with relatively few misclassifications. One of the most important results of this work is that the CAHFS managed to map 22 to 16 variables and maintain a competitive level of classification results. The fewer features used, the less data will be processed by the detection system, and therefore, fewer resources will be consumed by the detection system, which may be beneficial in smart-home environments where computing and storage resources are limited. Overall, the results indicate that the use of correlation along with the feature importance from the Mutual Information and the Random Forest may be able to reduce the number of selected features without affecting the detection performance significantly. These results come from one dataset with one experimental setup, however, they should be interpreted accordingly and not as generalizable to other datasets and/or deployments. Future research should investigate the approach on other datasets and under a variety of experimental conditions; ideally with replication to better assess generalizability.

VII. FUTURE WORK

These signs are promising, but there are a number of issues that could be explored further. The first is validation on additional datasets and independent test environments. Testing of CAHFS on various smart-home networks, attack scenarios, and device configurations would give a better understanding of the effectiveness of the method in expanding the dataset included in this study. Another direction which is useful is adaptive feature selection. Network traffic varies according to the change of applicable devices, applications, and communications. A feature selection technique that is adaptive to the changing nature of traffic could make the system more appropriate for long-term operation. Further research is also needed for comparing CAHFS with a wider variety of models such as ensemble methods, lightweight neural networks, or deep-learning models. In addition to classification performance, processing time, memory consumption, and computing cost would be beneficial to consider to see if the framework can be effectively deployed on smart-home gateways and edge devices.

DATA AVAILABILITY STATEMENT

The dataset used in this study is publicly available at: <https://data.mendeley.com/datasets/x95b37z2vy/1>

REFERENCES

- [1] S. M. Almufti et al., "Intelligent Home IoT Devices: An Exploration Of Machine Learning-Based Networked Traffic Investigation," *Jur. Ilmh. Ilm. Ter. Un. Ja*, vol. 8, no. 1, pp. 1–10, May 2024, doi: 10.22437/jiituj.v8i1.32767.
- [2] G. Vardakis, G. Hatzivasilis, E. Koutsaki, and N. Papadakis, "Review of Smart-Home Security Using the Internet of Things," *Electronics*, vol. 13, no. 16, p. 3343, Jan. 2024, doi: 10.3390/electronics13163343.

- [3] L. S. H. R. D. Jayalakshmi, V. R. Vimal, and L. Kanthan Narayanan, "IoT-Driven Energy Consumption Optimization in Smart Homes," in 2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies, Mar. 2024, pp. 1–5. doi: 10.1109/TQCEBT59414.2024.10545254.
- [4] C. M. Torres-Hernandez, M. Garduño-Aparicio, and J. Rodriguez-Resendiz, "Smart Homes: A Meta-Study on Sense of Security and Home Automation," *Technologies*, vol. 13, no. 8, p. 320, Aug. 2025, doi: 10.3390/technologies13080320.
- [5] L. Diana, P. Dini, and D. Paolini, "Overview on Intrusion Detection Systems for Computers Networking Security," *Computers*, vol. 14, no. 3, p. 87, Mar. 2025, doi: 10.3390/computers14030087.
- [6] P. Waghmode, M. Kanumuri, H. El-Ocla, and T. Boyle, "Intrusion detection system based on machine learning using least square support vector machine," *Sci Rep*, vol. 15, no. 1, p. 12066, Apr. 2025, doi: 10.1038/s41598-025-95621-7.
- [7] A. Thaljaoui, "Intelligent network intrusion detection system using optimized deep CNN-LSTM with UNSW-NB15," *Int. j. inf. tecnol.*, Feb. 2025, doi: 10.1007/s41870-025-02416-0.
- [8] A. K. B. Arnob, R. R. Chowdhury, N. A. Chaiti, S. Saha, and A. Roy, "A comprehensive systematic review of intrusion detection systems: emerging techniques, challenges, and future research directions," *Journal of Edge Computing*, vol. 4, no. 1, pp. 73–104, May 2025, doi: 10.55056/jec.885.
- [9] M. L. Ali, K. Thakur, S. Schmeelk, J. DeBello, and D. Dragos, "Deep Learning vs. Machine Learning for Intrusion Detection in Computer Networks: A Comparative Study," *Applied Sciences*, vol. 15, no. 4, p. 1903, Jan. 2025, doi: 10.3390/app15041903.
- [10] A. Kadam and P. S. Lokhande, "Design of an Intrusion Detection Model for Iot-enabled Smart Home," May 19, 2024, Social Science Research Network, Rochester, NY: 4833390. doi: 10.2139/ssrn.4833390.
- [11] K. Moinuddin and P. S., "An integrated machine learning and deep learning framework for intrusion detection in IoT smart homes," *Discov Internet Things*, Jun. 2026, doi: 10.1007/s43926-026-00390-y.
- [12] A. Abdusalomov, D. Kilichev, R. Nasimov, I. Rakhmatullayev, and Y. Im Cho, "Optimizing Smart Home Intrusion Detection With Harmony-Enhanced Extra Trees," *IEEE Access*, vol. 12, pp. 117761–117786, 2024, doi: 10.1109/ACCESS.2024.3422999.
- [13] H. H. Ali, J. R. Naif, and W. R. Humood, "A New Smart Home Intruder Detection System Based on Deep Learning," *Al-Mustansiriyah Journal of Science*, vol. 34, no. 2, pp. 60–69, Jun. 2023, doi: 10.23851/mjs.v34i2.1267.
- [14] U. H. Garba, A. N. Toosi, M. F. Pasha, and S. Khan, "SDN-based detection and mitigation of DDoS attacks on smart homes," *Computer Communications*, vol. 221, pp. 29–41, May 2024, doi: 10.1016/j.comcom.2024.04.001.
- [15] S. Batool, M. K. Abid, M. A. Salahuddin, Y. Aziz, and N. Aslam, "Integrating IoT and Machine Learning to Provide Intelligent Security in Smart Homes," vol. 07, no. 01.
- [16] M. M. Rahman, D. Gupta, S. Bhatt, S. Shokouhmand, and M. Faezipour, "A Comprehensive Review of Machine Learning Approaches for Anomaly Detection in Smart Homes: Experimental Analysis and Future Directions," *Future Internet*, vol. 16, no. 4, p. 139, Apr. 2024, doi: 10.3390/fi16040139.
- [17] P. Sadotra, P. Chouksey, M. Chopra, G. Thakur, and M. H. Nayak, "Intrusion Detection in Smart Homes: A Comprehensive Review," in 2024 International Conference on Progressive Innovations in Intelligent Systems and Data Science (ICPIDS), Dec. 2024, pp. 55–59. doi: 10.1109/ICPIDS65698.2024.00018.
- [18] B. Nair and V. Das, "Smart Home Intrusion Detection Dataset," vol. 1, Jan. 2026, doi: 10.17632/x95b37z2vy.1.
- [19] V. Kandasamy and A. A. Roseline, "Harnessing advanced hybrid deep learning model for real-time detection and prevention of man-in-the-middle cyber attacks," *Sci Rep*, vol. 15, no. 1, p. 1697, Jan. 2025, doi: 10.1038/s41598-025-85547-5.
- [20] A. G. Ayad, N. A. Sakr, and N. A. Hikal, "Fog-empowered anomaly detection in IoT networks using one-class asymmetric stacked autoencoder," *Cluster Comput*, vol. 28, no. 8, p. 550, Aug. 2025, doi: 10.1007/s10586-025-05234-y.
- [21] H. Hakami, M. Faheem, and M. Bashir Ahmad, "Machine Learning Techniques for Enhanced Intrusion Detection in IoT Security," *IEEE Access*, vol. 13, pp. 31140–31158, 2025, doi: 10.1109/ACCESS.2025.3542227.
- [22] A. A. Abu-Shareha and A. Amer, "Improving linear support vector machine classifier using mutual information feature weighting for intrusion detection system," in Seventh International Conference on Image, Video Processing, and Artificial Intelligence (IVPAI 2025), SPIE, Aug. 2025, p. 231. doi: 10.1117/12.3075315.
- [23] M. T. Abdelaziz et al., "Enhancing Network Threat Detection with Random Forest-Based NIDS and Permutation Feature Importance," *J Netw Syst Manage*, vol. 33, no. 1, p. 2, Oct. 2024, doi: 10.1007/s10922-024-09874-0.
- [24] B. Williams and L. Qian, "Semi-Supervised Learning for Intrusion Detection in Large Computer Networks," *Applied Sciences*, vol. 15, no. 11, p. 5930, Jan. 2025, doi: 10.3390/app15115930.
- [25] P. M. S. Makhdooni et al., "Network-based intrusion detection: a comparative analysis of machine learning approaches for improved security," *Journal of Cyber Security Technology*, vol. 10, no. 1, pp. 1–28, Jan. 2025, doi: 10.1080/23742917.2024.2447119.
- [26] F. A. Rafrastara, G. F. Shidik, W. Khozi, N. Rijati, and O. Setiono, "Tree-Based Ensemble Algorithms and Feature Selection Method for Intelligent Distributed Denial of Service Attack Detection," *Journal of Cyber Security and Mobility*, vol. 14, no. 1, pp. 1–24, Jan. 2025, doi: 10.13052/jcsm2245-1439.1411.
- [27] S. Priyadarshini, J. Nayak, P. B. Dash, R. Tanty, and S. Pattanaik, "Intelligent Intrusion Detection for Unmanned Aerial Vehicles to Enhance Defense Security System: An Improved Ensemble Learning Approach," in 2025 7th International Conference on Information Systems and Computer Networks (ISCON), Sep. 2025, pp. 1–6. doi: 10.1109/ISCON65210.2025.11341301.
- [28] D. K. Hariyani, Y. R. Reddy, K. Dave, T. Patel, D. Vekariya, and P. Khan, "Intrusion detection system: a python-based approach for network security," in Parul University International Conference on Engineering and Technology 2025 (PiCET 2025), May 2025, pp. 162–169. doi: 10.1049/icp.2025.1290.