Security and Privacy

WILEY

RESEARCH ARTICLE OPEN ACCESS

PCM-RF a Hybrid Feature Selection Mechanism for Intrusion Detection System in IoT

Naveed Ahmed[1] | Md Asri Ngadi[2] | Muhammad Siraj Rathore[3] | Azhar Mahmood[4]

[1]Pervasive Computing Research Group, Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Johor, Malaysia | [2]Department of Computer Science, Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Johor, Malaysia | [3]Department of Computer Science, Faculty of Computing, Capital University of Science and Technology Islamabad, Islamabad, Pakistan | [4]Department of Computer Science and Digital Technologies, School of Architecture, Computing and Engineering, University of East London, London, UK

Correspondence: Azhar Mahmood (a.mahmood3@uel.ac.uk)

Received: 1 October 2024 | **Revised:** 17 December 2024 | **Accepted:** 24 December 2024

Funding: The authors received no specific funding for this work.

Keywords: feature selection | Internet of Things and classification | Pearson correlation matrix | random forest

ABSTRACT

The integrity and security of data must be protected in the framework of the Internet of Things (IoT). This article addresses the difficulties presented by possible cyber threats to IoT devices by introducing a unique feature selection technique called Pearson correlation matrix with random forest (PCM-RF). IoT device security is greatly influenced by machine learning techniques, which mostly depend on the caliber of characteristics taken from IoT datasets. By merging the advantages of RF with PCM, PCM-RF maximizes feature selection and fine-tunes features to improve the efficacy of current machine learning techniques and bolster classification algorithms' detecting powers. The study focuses on addressing limitations in existing ways of training and testing classification algorithms, which frequently lack strategies for optimizing and fine-tuning features. Thirty-four different forms of network assaults are included in the IoTCIC2023 dataset, used to assess PCM-RF. Results demonstrate PCM-RF's effectiveness, with XGBoost achieving an astounding accuracy of 99.39% and an 86% detection rate. Comparative studies highlight PCM-RF's superiority in detection and classification results, offering insights into IoT security and emphasizing its potential to improve the overall device robustness in the IoT ecosystem.

1 | Introduction

IoT devices rely on a variety of connectivity choices to send and receive data, respond to commands, and function properly. Thus, maintaining the data safety is essential and pivotal toward the success of IoT devices. For this reason, many methodologies have been introduced and implemented in the field of machine learning. The machine learning methods performance solely based on the features extracted from IoT dataset. Therefore, a robust and optimized feature selection procedure is crucial to detect multiple attacks, and detecting novel attacks is necessary to mitigate the threat posed by the IoT devices.

Feature selection is an important data preparation approach used to identify the most relevant, applicable, and substantial feature space. To depict a record in a dataset for predictive modeling, it entails choosing a subset of the most distinctive and pertinent aspects from a vast collection of features [1]. This is a feature engineering technique where a dataset's item and attribute is used to lower the dimensionality of the issue at hand and speed up the classification process. Dimension reduction in a massive multidimensional dataset is the main driving force behind the feature selection process [2].

Feature selection is primarily difficult because the source dataset, which sometimes has a lot of characteristics, makes it difficult to choose only a few features. When working with a huge dataset, it can be challenging to identify precise links and draw conclusions since certain aspects are closely connected to the issue at hand while other variables are not. The selection result would change if every feature was chosen. As a result, choosing the qualities that are most pertinent to the particular situation at hand is crucial to coming up with the optimal solution [2].

In the feature selection process, it is best to stay away from any aspect that might have an impact on the conclusion, produce false findings, or take a lot of time to analyze. When irrelevant characteristics are present in the data, the models' accuracy might suffer and the model may start to learn from these features. Consequently, a subset of the original dataset is produced by removing characteristics that are not important. This can be done manually or automatically.

Existing feature selection approaches in intrusion detection systems often do not incorporate techniques for fine-tuning selected features. This absence of fine-tuning may limit the performance of the classification models during training and testing, thereby not utilizing the full potential of the data. It appears that many existing methods may not fully utilize feature fine-tuning techniques, which could impact their ability to achieve optimal performance. This observation suggests that there may be opportunities to enhance these methods. The increasing complexity and volume of data from IoT devices require advanced feature selection techniques to ensure effective and secure data processing. Traditional methods often struggle with high-dimensional datasets, nonlinearity, and noise, which can hinder model performance.

In this article, we propose a novel feature selection approach that combines the Pearson correlation matrix (PCM) with random forest (RF) to enhance feature selection from the dataset. The use of PCM allows us to identify and filter out highly correlated features, which helps in reducing redundancy and improving computational efficiency. RF, on the other hand, evaluates the importance of the remaining features, ensuring that only the most relevant ones are selected for model training. This hybrid approach is compelling because PCM addresses the issue of feature correlation, which is crucial for handling high-dimensional data, while RF provides a robust assessment of feature relevance, contributing to improved model performance. Combining these methods leverages the strengths of both techniques: PCM's capability to reduce redundancy and RF's ability to rank feature importance, thus creating a more effective and efficient feature selection mechanism than using either method alone. For feature evaluation, feature vectors are fed to multiple machine learning techniques encompassing XGBoost, multilayer perceptron, naïve Bayes, logistic regression, decision tree, KNN, and majority voting. Our approach significantly improves feature selection by integrating linear and nonlinear pattern recognition, thus overcoming the limitations of traditional methods. The results show that our method achieves an accuracy of 99.39% with XGBoost, representing a notable improvement over existing solutions. For instance, traditional methods typically achieve accuracies around 98%, indicating a performance boost of approximately 1.39 percentage points with our approach. Additionally, other models evaluated also show substantial improvements compared to conventional techniques. This section presents a comprehensive comparison of our method with existing techniques, highlighting its superior accuracy and robustness in handling complex IoT datasets.

This article is organized as follows: In Section 2, we present a comprehensive literature review that discusses related work and identifies gaps in current feature selection techniques. Section 3 covers the design and implementation of our proposed method, including the organization of the dataset, detailed descriptions of the feature selection methods used, and an analysis of the results. This analysis includes a comparison of our approach with existing state-of-the-art methods. Section 4 provides a thorough presentation of the results and their analysis, highlighting the performance improvements achieved by our method. Finally, Section 5 offers a summary of the findings and discusses potential directions for future research.

2 | Literature Review

2.1 | Overview of Traditional Feature Selection Methods

Traditional feature selection approaches are critical in improving the accuracy of machine learning algorithms by detecting and choosing the most relevant characteristics from a dataset. These techniques are crucial for lessening the effects of the "curse of dimensionality," which is the condition in which there are more features than samples, which causes overfitting and lower model performance. Traditional techniques increase model interpretation, computational speed, and enhanced generalization to new data by focusing on the most relevant aspects. Applying these methods wisely is essential to maximizing model performance in a variety of machine learning applications and navigating the intricate world of feature spaces. To minimize noise and duplication in the dataset, popular approaches are included that allow valuable information to be extracted which is shown in Figure 1.

2.1.1 | Filter Method

Filter methods analyze each feature's inherent properties without considering target variable interaction. Common metrics include correlation, statistical testing, and information gain. For example, the RPFMI filter-based method effectively selects intrusion detection characteristics by balancing redundancy, classifier connection, and class label correlation. These approaches are computationally efficient and can be utilized before model training. They may neglect feature interactions and dependencies, resulting in suboptimal feature subsets [3]. In contrast, hybrid feature selection methods combine multiple approaches to leverage the strengths of each.

The ineffective penalty has been introduced between the selected feature shared data algorithm (RPFMI), a filter-based feature

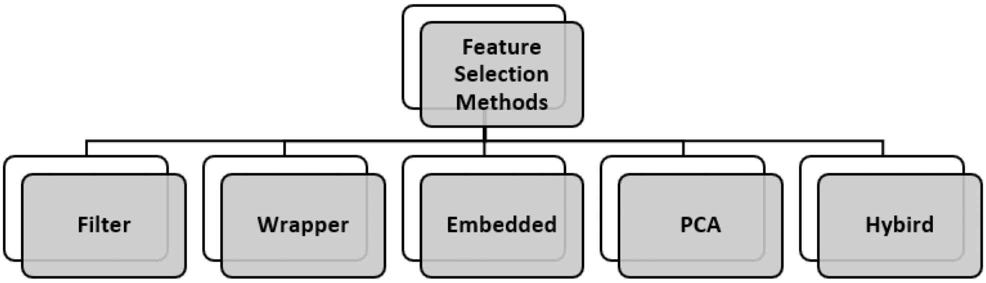


FIGURE 1 | Classification of feature selection techniques.

selection process, to identify the best features in terms of redundant function, classifier connection, class label correlation, and limited data samples [4]. The research's high-accuracy experiments showed that the proposed RPFMI chooses the optimal intrusion detection characteristics. Another notable approach, dynamic-change of selected feature (DCSF), uses conditional mutual information to select the most informative features, leading to improved precision. This technique improves the relevance of selected features by accounting for dynamic data changes, resulting in more precise and effective results [5]. This new method uses conditional mutual data between contestant features and labeled classes to find the most informative features, unlike other filter techniques that use mutual information to calculate candidate feature relevance to the selected optimum feature subset. Experimentally, DCSF exceeds all other classification methods in average precision [5].

2.1.2 | Wrapper Method

Wrapper methods evaluate subsets of features based on model performance. The algorithm must be tested and trained with different feature subsets regularly. Wrapper techniques capture feature dependencies more accurately, improving feature significance evaluation. For instance, Boolean-Particle-Swarm Optimization (BoPSO) enhances feature selection for classifying diseases with the use of a support vector machine (SVM). They overfit and are computationally expensive for large datasets [6]. Significantly, Ron Kohavi developed wrapper techniques. His contributions to machine learning evaluation and feature selection are notable. Wrapper approaches for model-specific feature selection and feature subset evaluation with regard to the desired learning method are frequent themes in his research [7].

A wrapper-based feature selection strategy based on Boolean-Particle-Swarm Optimization (BoPSO) has been developed to improve hepatic and renal illness classification [8]. Abdominal CT image slices were used to derive first- and second-order statistical properties from the Gray level-co-occurrence-matrix (GLCM). Two new updated BPSO algorithms, velocity-bounded-BoPSO (VbBoPSO) and improved-VbBoPSO, were proposed employing an SVM classifier fitness function. IVbBoPSO's top features gave the PNN/SVM liver sickness precision of 77.14% and 82.86%. However, employing elite features picked by VbBoPSO, PNN/SVM attained renal disease accuracies of 77.17% and 90.3% [8].

Major developments in wrapper strategies focus on model assessment and feature selection. His study shows how wrapper techniques improve predictive model accuracy by selecting key characteristics [9]. Their research examines the intricate interaction between feature subsets and model accuracy, noting that feature relevance directly affects machine learning model success. By pushing wrapper approaches, which evaluate feature subsets through particular models, Liu made substantial contributions to model evaluation and feature selection. These methodologies provide useful insights for scholars and practitioners seeking predictive modeling optimization [9]. His wrapper techniques to identify the most significant features are embedded in machine learning and statistical modeling.

"Feature selection method for clustering" advances unsupervised learning, especially feature selection wrapper approaches. This article examines how feature selection strategies, often used in wrapper approaches, affect supervised learning tasks like dimensionality reduction and clustering. The study reveals that proper feature selection improves unsupervised techniques' precision and efficacy slightly but significantly. This study emphasizes wrapper strategies in unsupervised events to improve dimensionality reduction and clustering accuracy [10]. This helps readers understand how feature selection affects unsupervised learning.

2.1.3 | Embedded Method

Feature selection is incorporated into the model training procedure using embedded approaches. These techniques pick features as the model is being constructed. Throughout the model training phase, embedded approaches can capture feature interdependence and are computationally effective. Their performance in capturing intricate feature interactions can be inferior to that of wrapper approaches, and their selection criteria are frequently model-specific [11]. Feature selection is a part of the learning algorithm in embedded systems. Once the learning algorithm's training is finished, the feature subset may be acquired. The filter approach and the embedding method are comparable. It differs from the filter technique in that it uses model training to compute the feature score. This method's fundamental idea is to identify the model by choosing features that are crucial for model training [12]. In the meanwhile, the embedded approach represents a balance between the filter & wrapper approaches. The accuracy of classification achieved by the embedded approach is higher than that of the filter method. Embedded techniques are more prone to the overfitting problem & have a lower complexity of algorithms when compared to wrapper methods [13].

Using as few features as feasible, embedded feature selection approach was created a separation plane to discriminate between two distinct point sets in an n-dimensional feature space. It is based on concave minimization and SVM. This technique maximizes the distance between the two boundary planes of the separation plane and reduces the weighted mean of the distances between the boundary plane and the erroneous classification sites. Regression methods for learning are a common foundation for embedded techniques [14]. To eliminate outliers, an effective and robust feature selection technique has been suggested based on L2;1 norm. This approach uses regularization and joint L2;1-norm minimization on the loss function, and it suggests a useful solution to handle a number of joint L2;1-norm minimization issues [15]. Unsupervised discriminative feature selection (UDFS) is a feature selection technique that was proposed. UDFS makes use of discriminative data and the local structure of the data dispersion to optimize an L2;1-norm standardized reduction loss function [16]. Utilizing the training subspace as an intermediary space, Liang et al. introduced a novel resistant structuralized subspace learning (RSSL) technique [17].

2.1.4 | PCA Method

Principal component analysis (PCA) converts features into uncorrelated variables called principle components to reduce dimensionality and maintain the most critical information. This helps de-redundancy and simplifies complex datasets by emphasizing key trends. In linear connections with strong correlation, PCA removes multicollinearity. Robust PCA, developed by Ahmadi, addresses this limitation by handling datasets with anomalies and noise.

Its linear assumptions limit its applicability to complex nonlinear datasets. Due to nonlinear relationships between variables, PCA may not accurately detect the underlying patterns, resulting in less-than-ideal representations. Thus, while PCA can reveal the structure of linearly connected data, its usage with nonlinear datasets should be carefully considered.

Pioneering statistician introduced PCA in 1901, laying the groundwork for its mathematical structure. By establishing eigenvectors and eigenvalues, two key mathematical elements of PCA, they advanced the subject. Eigenvalues measure the size of this fluctuation, while eigenvectors show the data's most significant differences. These principles allowed Pearson to methodically convert correlated variables into uncorrelated variables, or main components [18]. By enabling feature selection, dimensionality reduction, and multivariate data analytics, this ground-breaking mathematical formulation made PCA a popular machine learning and statistical analysis tool.

Eigenanalysis was introduced, which helped establish PCA theory [19]. He enhanced his analytic understanding of PCA by investigating eigenvalues and eigenvectors, building on Karl Pearson's ground-breaking work. Eigenanalysis, his term, helped explain dataset relationships. He made PCA a powerful multivariate statistical analysis tool. His work increased PCA's use and mathematical rigor in statistical and data analysis [20].

Robust PCA, notably for dataset anomalies and outliers, has improved. His study focuses on generating powerful PCA variants that can resist odd or noisy data. Ahmadi's robust PCA algorithms help identify features in datasets with outliers that may affect standard PCA. Because they are robust, these PCA versions can uncover and keep essential features even with anomalous data sets. This makes feature selection approaches more reliable in outlier-resistant modeling and data preparation [21]. Ahmadi's advances make PCA more durable and versatile for complicated real-world datasets with anomalies.

Chiefly in feature selection, Ghosh & Mandal have advanced PCA. His study emphasizes the importance of detailed preprocessing and scaling when using PCA for feature selection. PCA needs proper preprocessing to capture feature variance. Including data normalization and scaling [22]. Their research illuminates these preprocessing steps and offers tips for improving PCA-based feature selection [22]. His study shows that PCA is reliable, strong, and can extract valuable data for feature selection from various datasets, boosting its application.

2.1.5 | Hybrid Method

The hybrid feature selection process, which combines RF and PCM, is a complete strategy that takes use of the advantages of these two unique approaches. For example, recent studies have introduced advanced feature selection techniques specifically designed for network intrusion detection in IoT environments. The PCM offers insights into feature interdependencies by making it easier to identify linear correlations between variables. Combining this with the stable ensemble learning algorithm RF improves the mechanism's capacity to recognize both linear and nonlinear patterns, leading to a more intricate and thorough feature selection procedure. RF is an excellent option for managing nonlinear complexity, whereas PCM is best suited for linear interactions. By combining the interpretability of correlation-based selection with the predictive capability of RF, the hybrid technique aims to strike a balance and produces a feature selection approach that is more adaptable and durable [23]. Through the combination of various strategies, the mechanism seeks to address the shortcomings of each method separately, providing a comprehensive solution that can extract pertinent features from a variety of datasets and enhance machine learning models' overall performance.

The groundwork was created for the RF method, which is a major component of the hybrid feature selection process. His ground-breaking work in decision trees and ensemble learning transformed predictive modeling and laid the groundwork for RF's capacity to deal with large datasets and capture sophisticated feature interactions. Jerbi and Brahim's efforts have had a major impact on the area, and his influence on improving machine learning techniques is seen in the use of RF in hybrid feature selection algorithms [24]. The algorithm's resilience and high-dimensional data handling capabilities have an impact on how the algorithm functions in the hybrid feature selection process. The practical use of RF in identifying both linear and nonlinear correlations has been affected in machine learning and statistical techniques [25]. This has improved the hybrid mechanism's ability to choose important features across a range of datasets. We hypertuned the RF in our technology for better performance and accuracy.

Our novel filter method emphasizes features' true ranks indicated by ReliefF and Fisher Score rather than reciprocal redundancy.

Mutual Information, ReliefF, and Fisher Score (MIRFFS) uses differential evolution (DE) to search. One technique addresses single-objective problems and the other addresses multiobjective problems [26]. An innovative filter-based feature selection technique termed multivariate-relative discrimination criteria (MRDC) improves text classification accuracy [27]. Minimal-redundancy and maximal-relevancy (mRmR) reduce feature space dimensionality. MRDC determines relevant features using a comparative discrimination criterion, or RDC [28]. The correlation coefficient matrix helps RDC find features it cannot classify. By integrating three trustworthy filter procedures, a unique feature selection technique (vectors of scores/V-score) maximizes benefits and minimizes drawbacks to uncover useful dataset characteristics [29]. They improved prediction results by stabilizing feature ranking scores with data gain, chi-square statistics, and intercorrelation approaches (CFS) [29].

In summary, our contribution refines feature selection by integrating advanced techniques to enhance traditional methods. We leverage ReliefF and Fisher Score to focus on true feature relevance, and employ MIRFFS with DE to address both single-objective and multiobjective problems. Additionally, we introduce the MRDC and minimal-redundancy maximal-relevancy (mRmR) techniques for improved dimensionality reduction and feature relevance. Our novel V-score approach combines data gain, chi-square statistics, and correlation measures to stabilize and enhance feature ranking, offering a more robust and effective feature selection framework.

2.2 | Recent Advances in Feature Selection for Intrusion Detection

Recent research has explored innovative approaches to enhance feature selection in network intrusion detection, particularly in IoT environments. Probabilistic Dependency Trees and Evolutionary Algorithms (2022): This approach enhances network intrusion detection by selecting effective features based on probabilistic dependency trees. The integration of evolutionary algorithms significantly improves detection accuracy and reduces computational costs.

RF and PSO Algorithm (2021): By combining RF with particle swarm optimization (PSO), this method enhances intrusion detection in IoT systems. The RF efficiently handles nonlinear dependencies, while PSO optimizes feature selection, leading to higher detection rates.

SVM for intrusion detection (2021): The use of SVMs, combined with optimized feature subsets, results in improved classification performance for network intrusion detection.

Hybrid PSO-logistic regression algorithm (2022): This hybrid method integrates PSO with logistic regression to model intrusion detection behavior, offering superior performance compared to traditional models.

While the studies cited provide valuable contributions to intrusion detection in IoT environments, they generally lack a focus on advanced feature selection techniques that are specifically tailored for high-dimensional IoT datasets. Most rely on traditional methods that may not fully address the complexity of IoT traffic or the resource constraints of IoT devices. This paper addresses these gaps by proposing the PCM-RF (PCM with RF) feature selection technique, which improves both detection accuracy and computational efficiency, offering a more refined solution to the challenges of IoT security.

Recent advancements in IoT security emphasize the importance of sophisticated detection methods that combine feature selection and machine learning for improved accuracy and adaptability. For example, approaches using probabilistic dependency trees and evolutionary algorithms have shown promise in feature selection, contributing to more robust intrusion detection systems [30]. Enhancements like the integration of RF and PSO algorithms further demonstrate how hybrid methods can optimize detection capabilities and reduce false positives in IoT environments [31]. The application of support vector machines to intrusion detection, another effective method, highlights the trend toward leveraging machine learning for security in network systems [32]. Additionally, the use of hybrid algorithms such as PSO combined with logistic regression shows a potential pathway for PCM-RF to evolve, particularly in terms of adaptive feature selection in dynamic IoT networks [33].

Authentication and privacy-aware frameworks also play a critical role. Methods that incorporate behavioral biometrics, for instance, extend beyond traditional liveness checks, offering greater resilience against attacks [34]. Similarly, privacy-aware frameworks like split learning facilitate data security without sacrificing functionality in networked environments [35]. These insights reinforce PCM-RF's objectives to enhance IoT security through dynamic feature selection and privacy-conscious design. Incorporating gesture-based authentication, as explored in recent studies, could also align with PCM-RF's potential applications, especially in scenarios requiring interaction with user behavior [36]. Finally, advancements in web application firewalls (WAFs) demonstrate the need for systems to adapt to evolving threats, a concept PCM-RF embodies through its adaptive feature selection mechanism [37].

2.3 | Limitations of Individual Methods in Handling Complex Datasets

The traditional feature selection techniques have their limits, especially when working with big datasets, even if they have shown their value in a number of applications.

2.3.1 | Curse of Dimensionality

The curse of dimensionality provides a tremendous obstacle for many classic feature selection approaches, resulting in an imbalance in which the number of features considerably outnumbers the available samples in a dataset. When models struggle to generalize patterns from limited information, this phenomenon leads to overfitting, which impairs performance and may result in predictions that are not entirely accurate. Techniques that depend on having a large number of samples for every feature are less effective in high-dimensional areas when dimensionality is a problem [38]. Traditional feature selection methods have distinct pros and cons. Filter methods are efficient but may miss complex patterns [39]. Wrapper methods capture intricate feature interactions but are computationally expensive. Embedded methods offer a balance between efficiency and accuracy but are model-specific. Dimensionality reduction techniques like PCA reduce feature space effectively but may struggle with nonlinear relationships and interpretability. Choosing the right method depends on the dataset and problem requirements.

2.3.2 | Nonlinearity

When working with complex datasets, linear approaches like PCA and filter techniques face difficulties due to the limitations of nonlinearity. Although these techniques are quite good at capturing linear correlations, they are not as good at deciphering the complex nonlinear patterns that are present in many real-world situations. The efficacy of linear approaches is hampered by their inability to identify and capture subtle nonlinear interactions in the data. This constraint is most noticeable in instances where complex, nonlinear transformations describe the interactions between variables [40]. In certain situations, linear approaches could miss important details, producing less-than-ideal representations and thus jeopardizing the correctness of the models. Because nonlinear structures are frequently present in complicated datasets, it is critical to develop more advanced feature selection techniques that can capture nonlinear relationships to guarantee the accuracy and resilience of machine learning models in a variety of applications.

2.3.3 | Interactions and Dependencies

By using a particular machine learning model to evaluate subsets of features, wrapper techniques are highly praised for their ability to capture complex feature relationships. Their effectiveness does, however, come at a price, since they are often computationally costly and may not scale well to datasets with a high number of characteristics or occurrences. Because wrapper approaches are iterative, requiring repeated cycles of model training and assessment for various feature subsets, there is a computational cost associated with them. When working with large datasets, this repetitive procedure can become unreasonably resource-intensive, which restricts the use of wrapper approaches in situations when computing efficiency is essential? Wrapper approaches for feature selection in large-scale datasets require careful evaluation of the trade-off between efficiency and scalability due to their computing demands, which can be challenging even if they can unravel complicated feature connections [41].

2.3.4 | Inability to Handle Noise

Traditional feature selection techniques may be sensitive to characteristics that are irrelevant or noisy, which reduces their ability to identify variables that are actually meaningful even in the presence of faulty data. These techniques can be misled by the existence of noise, which is defined as random or unimportant oscillations in the data. This could compromise the robustness of the variables that are chosen and result in the inclusion of less important aspects [42]. The accuracy and dependability of the chosen feature subsets may be impacted in situations when datasets contain noisy components or irrelevant features, making it difficult for traditional approaches to distinguish between important signals and random changes. Noise-resistant or filtering strategies must be incorporated into the method for choosing features in order to overcome this constraint and make sure that the variables that are found really add to the predictive potential of machine learning algorithms.

To address limitations such as the curse of dimensionality, nonlinearity, and high computational costs, our approach employs a hybrid feature selection method combining PCM for identifying linear correlations and RF for capturing nonlinear interactions. PCM filters out irrelevant features based on linear relationships, while RF evaluates feature importance, effectively managing both linear and nonlinear dependencies. This integration allows our method to handle complex datasets more effectively, improve feature relevance, and reduce noise, leading to more accurate and scalable feature selection compared to traditional techniques.

Despite the significant progress in feature selection techniques for intrusion detection, gaps remain in handling the complexity and scalability of IoT environments. While many studies focus on individual methods, there is a lack of comprehensive hybrid models that integrate different feature selection techniques (e.g., combining filter and wrapper methods) to handle both linear and nonlinear patterns effectively. Current methods often struggle with the high-dimensionality and real-time constraints of IoT networks. More efficient and scalable techniques are needed to address these challenges.

2.3.5 | Comparison

The performance comparison summary of our hybrid PCM-RF feature selection approach with the existing traditional approaches is presented in Table 1. However, to provide a more comprehensive evaluation of the PCM-RF method's effectiveness, it would be beneficial to compare its performance with additional algorithms, such as convolutional neural networks (CNNs) and other advanced machine learning approaches commonly applied in IoT security. CNNs, for instance, have demonstrated strong adaptability in capturing complex data patterns, which are essential for detecting sophisticated attack signatures in IoT environments. By including CNN and newer methods—such as Transformer-based architectures or deep ensemble models—the paper could present a more robust benchmark, showcasing PCM-RF's performance not only against traditional machine learning algorithms but also alongside these contemporary techniques.

Incorporating these additional algorithms would allow for a deeper analysis of PCM-RF's accuracy, detection rate, and computational efficiency in relation to state-of-the-art approaches. Furthermore, highlighting PCM-RF's strengths in feature selection and model interpretability, especially in contrast to the often "black-box" nature of deep learning models, would underscore its practical advantages for IoT applications. This comparative analysis would enrich the study by positioning PCM-RF as a practical, resource-efficient choice, thus enhancing its relevance for real-world IoT security scenarios where both performance and transparency are critical.

TABLE 1 | Comparison of feature selection methods.

Method	Key features	Strengths	Limitations	Performance
Filter methods	Correlation, statistical tests, information gain	Computationally efficient, premodel training	May miss feature interactions, suboptimal feature subsets	98% accuracy (typical)
Wrapper methods	Subset evaluation based on model performance	Captures feature dependencies	Computationally expensive, prone to overfitting	98.5% accuracy (typical)
Embedded methods	Feature selection integrated into model training	Captures feature interactions, computationally effective	Model-specific, may overfit	98.2% accuracy (typical)
PCA	Converts features to principal components	Reduces dimensionality, simplifies datasets	Linear assumptions, struggles with nonlinear data	97% accuracy (typical)
Hybrid methods (e.g., PCM-RF)	Combines techniques such as PCM and RF	Integrates linear and nonlinear pattern recognition	Complexity of combining methods	99.39% accuracy

3 | Design and Implementation of Proposed PCM-RF Approach

This section discusses the implementation of the proposed hybrid PCM and RF-based feature selection approach. The proposed PCM-RF approach is developed to enhance the performance of intrusion detection and classification system. The PCM uses correlation, similarity, interrelationships among data instances. To record the multivariate trends and continuous changes in the data flow, PCM approach is favored because of its abilities to adapt to dynamic and ever-evolving environments [43].

3.1 | Pearson Correlation Matrix

The following Equation (1) is used to calculate the PCM among the features.

$$r = \frac{\sum \left(X_i - X\right)\left(Y_i - Y\right)}{\sqrt{\sum \left(X_i - X\right)^2 \sum \left(Y_i - Y\right)^2}} \tag{1}$$

where r is a correlation coefficient, X_i and Y_i are the values of x and y variables in the dataset, respectively. X and Y are the mean values of x and y variables in the dataset, respectively. Features with a correlation coefficient above a specified threshold (set at 0.9 in this study) are considered highly correlated, and one of the features is discarded. This threshold was selected to minimize feature redundancy and multicollinearity, ensuring that only distinct features are retained. In this phase, missing values are handled through interpolation before calculating correlations, ensuring data consistency.

PCM's ability to adapt to dynamic environments makes it particularly suitable for analyzing continuously changing data flows, such as in IoT networks. The correlation threshold was tuned through experiments, where values between 0.7 and 0.9 were evaluated, and 0.9 was found to offer the best trade-off between feature elimination and model performance.

3.2 | Random Forest

Once PCM has reduced the dataset to a more manageable feature vector, RF is employed for feature ranking. The RF model is initialized with 100 decision trees (a common starting point for hyperparameter tuning) and trained on the reduced feature set. RF computes the feature importance score for each feature, which indicates how much each feature contributes to the model's predictions.

The top 20 features are selected based on their importance scores, determined after hyperparameter tuning for RF. Grid search was used to fine-tune parameters such as the number of trees (evaluating values from 50 to 500) and the maximum depth of trees. This tuning ensures that the RF model does not overfit or underfit, providing an optimized selection of features.

3.3 | Integration of PCM and RF

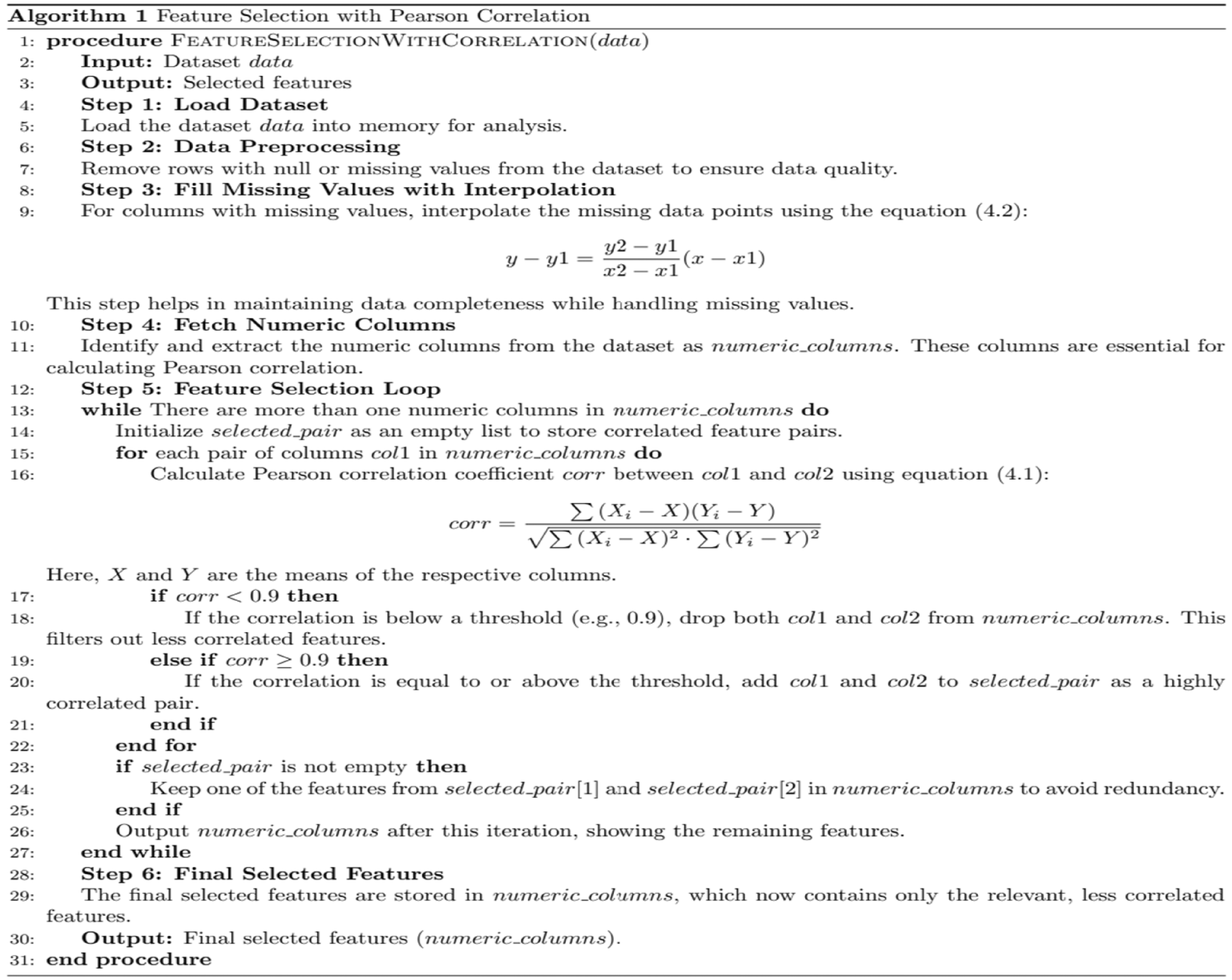
The integration of PCM and RF is a two-step process:

a. PCM removes redundant features: By calculating correlations and discarding highly correlated features, PCM ensures that only unique and nonredundant features are passed to the next phase.

b. RF ranks the remaining features: RF assigns importance scores to the remaining features, allowing for the selection of the most impactful features.

This hybrid approach leverages PCM's strength in filtering out redundant features and RF's ability to assess feature relevance. The combination of these methods creates a streamlined and efficient feature selection process that improves the performance of intrusion detection systems.

The pseudo code of PCM is given in Figure 2, that outlines Phase 1 of the feature selection process using PCA (PCM). It begins with loading and preprocessing the dataset to handle missing values



Algorithm 1 Feature Selection with Pearson Correlation

```
procedure FEATURESELECTIONWITHCORRELATION($data$)
    Input: Dataset $data$
    Output: Selected features
    Step 1: Load Dataset
    Load the dataset $data$ into memory for analysis.
    Step 2: Data Preprocessing
    Remove rows with null or missing values from the dataset to ensure data quality.
    Step 3: Fill Missing Values with Interpolation
    For columns with missing values, interpolate the missing data points using the equation (4.2):
        $y - y1 = \frac{y2 - y1}{x2 - x1}(x - x1)$
    This step helps in maintaining data completeness while handling missing values.
    Step 4: Fetch Numeric Columns
    Identify and extract the numeric columns from the dataset as $numeric\_columns$. These columns are essential for calculating Pearson correlation.
    Step 5: Feature Selection Loop
    while There are more than one numeric columns in $numeric\_columns$ do
        Initialize $selected\_pair$ as an empty list to store correlated feature pairs.
        for each pair of columns $col1$ in $numeric\_columns$ do
            Calculate Pearson correlation coefficient $corr$ between $col1$ and $col2$ using equation (4.1):
                $corr = \frac{\sum (X_i - X)(Y_i - Y)}{\sqrt{\sum (X_i - X)^2 \cdot \sum (Y_i - Y)^2}}$
            Here, $X$ and $Y$ are the means of the respective columns.
            if $corr < 0.9$ then
                If the correlation is below a threshold (e.g., 0.9), drop both $col1$ and $col2$ from $numeric\_columns$. This filters out less correlated features.
            else if $corr \geq 0.9$ then
                If the correlation is equal to or above the threshold, add $col1$ and $col2$ to $selected\_pair$ as a highly correlated pair.
            end if
        end for
        if $selected\_pair$ is not empty then
            Keep one of the features from $selected\_pair$[1] and $selected\_pair$[2] in $numeric\_columns$ to avoid redundancy.
        end if
        Output $numeric\_columns$ after this iteration, showing the remaining features.
    end while
    Step 6: Final Selected Features
    The final selected features are stored in $numeric\_columns$, which now contains only the relevant, less correlated features.
    Output: Final selected features ($numeric\_columns$).
end procedure
```

FIGURE 2 | Pseudocode of feature selection scheme Phase 1.

and prepare the data. The core step involves calculating feature correlations using Equation (1), which measures the linear relationship between feature pairs, as described in reference [43]. Features are evaluated against a correlation threshold of 0.9; pairs with correlations above this threshold are included in the feature vector, while those below are discarded. Additionally, Equation (2) is used for interpolating missing values during preprocessing. This approach ensures the feature set is streamlined and free of redundancy before moving to the next phase.

Figure 3 shows the phase 1 of feature selection process of PCM. For our study, we selected the IoTCIC2023 dataset due to its comprehensive and diverse range of features relevant to IoT security. The dataset is then loaded, and data preprocessing is carried out to ensure the data are clean and suitable for further analysis. The PCM selects a pair of features and then calculates correlation among them. Given the correlation threshold at 0.9, if the correlation is less than 0.9; one of the features is dropped from the pair. If it is greater than 0.9; both features are added to the feature vector. The 0.9 threshold is chosen to minimize redundancy and avoid multicollinearity by keeping only highly distinct features. This balance helps simplify the model while retaining valuable information. The feature vector obtained from PCM is further processed using RF. The RF is initialized and trained on the selected features. Then calculate the feature importance score. As a result, the top 20 features are nominated. Using PCM to reduce redundancy and RF to rank feature importance combines efficient feature selection with effective ranking, ensuring a streamlined and impactful model. Combining PCM and RF leverages their strengths: PCM efficiently reduces feature redundancy by selecting uncorrelated features, while RF evaluates feature importance to rank and refine them. This approach ensures a robust and effective feature selection process, optimizing model performance by first eliminating redundant features and then identifying the most influential ones. The flowchart of RF is shown in Figure 3.

The feature vector obtained from PCM is further processed using RF. The RF is initialized and trained on the selected features. Then calculate the feature importance score. As a result, the top 20 features are nominated. Using PCM to reduce redundancy and RF to rank feature importance combines efficient feature selection with effective ranking, ensuring a streamlined and impactful model. Pseudo code of RF important feature selection Phase 2 is given in Figure 4.

Figure 5 illustrates Phase 2 of the feature selection process using RF. It starts with the feature vector obtained from the PCM phase, which has reduced feature redundancy. RF is then initialized and trained on these features. After training, RF calculates the importance scores for each feature, reflecting their contribution to model performance. The features are ranked based on these importance scores, and the top-ranked features are selected for further use. This optimized feature set, consisting of the most significant features, is then used for building and refining the final model.

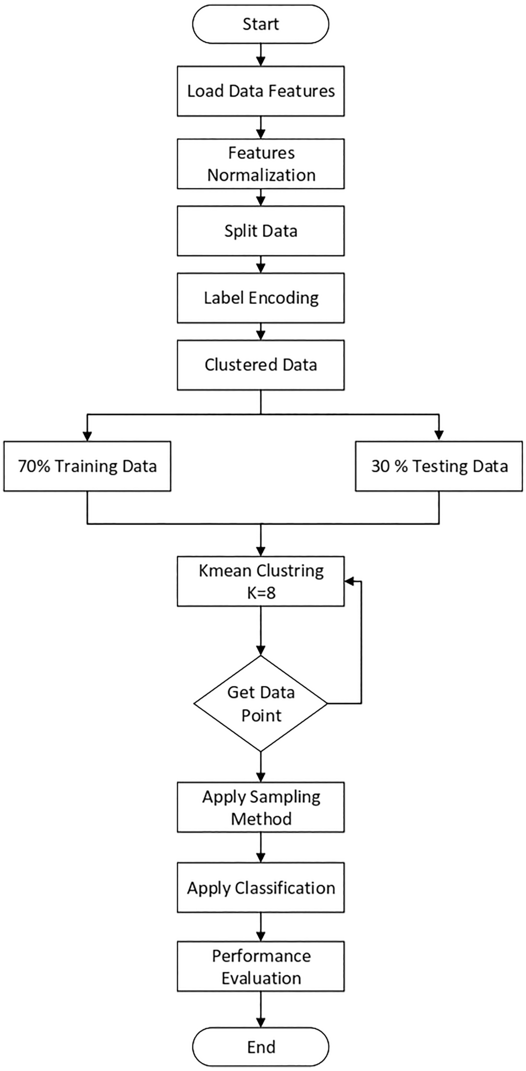


FIGURE 3 | Flowchart of RF.

This figure depicts the pseudocode for Phase 1 of the feature selection process, which involves PCA (PCM). It outlines the steps for calculating the correlation between feature pairs, applying the correlation threshold to decide whether to keep or discard features, and assembling the feature vector based on these correlations.

This flowchart illustrates the RF phase, which is Phase 2 of the feature selection process. It details how RF is initialized, trained on the features selected by PCM, and how it calculates feature importance scores. This phase ranks the features based on their contribution to the model's performance.

This figure provides the pseudocode for Phase 2, which involves using RF to process the feature vector obtained from PCM. It outlines the steps for training the RF model on these features, computing feature importance scores, and selecting the top features based on their importance.

This figure visually represents the steps and mechanisms involved in optimizing feature selection in Phase 2, complementing the pseudocode in Figure 5. It shows the process of ranking features with RF and refining the feature set.

Figure 6 illustrates how PCA (PCM) evaluates the correlation between features. The figure demonstrates the process of calculating and visualizing feature correlations to identify redundancy. PCM computes the correlation coefficient for each pair of features using Equation (1), which measures the strength and direction of their linear relationship. Features with high correlations (above the set threshold of 0.9) are considered redundant and one of the pair is discarded, while those with low correlations are retained. This visualization helps readers understand how PCM filters out redundant features by highlighting which features are kept or removed based on their correlation scores.

Algorithm 2 Random Forest Feature Importance

```
procedure RandomForestFeatureImportance(data)
    Input: Dataset data
    Output: Top 20 important features
    Step 1: Select Features from Pearson Correlation
    SelectFeatures(data)                                          ▷ Refer to Algorithm 1 for details
    Step 2: Initialize a Random Forest Classifier
    rf_model ← InitializeRandomForest                             ▷ Initialize a Random Forest classifier
    Step 3: Train the Random Forest Classifier
    TrainRandomForest(rf_model, data)                             ▷ Train the Random Forest classifier on selected features
    Step 4: Calculate Feature Importance Scores
    importance_scores ← CalculateFeatureImportance(rf_model)      ▷ Calculate feature importance scores
    Step 5: Select the Top 20 Features
    top_features ← SelectTopFeatures(importance_scores, 20)       ▷ Select the top 20 features
    Output: top_features                                          ▷ Top 20 important features
end procedure
```

FIGURE 4 | Pseudocode of optimized feature approach Phase 2.

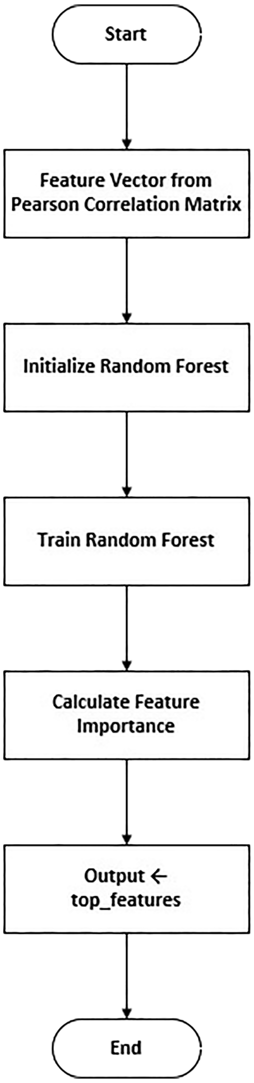

Start
Feature Vector from Pearson Correlation Matrix
Initialize Random Forest
Train Random Forest
Calculate Feature Importance
Output ← top_features
End

FIGURE 5 | Optimized feature mechanism Phase 2.

The goal is to reduce multicollinearity and simplify the feature set, making it more manageable and effective for further analysis. This approach ensures that only the most relevant and distinct features are included in the final model.

Figure 7 depicts the outcome of applying PCA (PCM) after feature dropout. This figure shows the feature set once redundant features have been removed based on the correlation threshold established in Phase 1. After evaluating feature correlations and applying the threshold, PCM eliminates features that were highly correlated with others, resulting in a streamlined and more focused feature set.

The figure visually represents the remaining features and highlights which features have been dropped. This postdropout feature set is now less redundant and more distinct, providing a cleaner basis for the subsequent RF phase. The purpose of this figure is to illustrate how PCM reduces feature redundancy, ensuring that only the most informative and nonredundant features are retained for further processing.

Figure 8 illustrates the feature importance scores computed by the RF algorithm. After PCM has filtered out redundant features, RF is applied to evaluate the importance of each remaining feature in predicting the target variable. The figure displays the importance scores assigned to each feature, with higher scores indicating greater contributions to the model's performance. This ranking is crucial for identifying which features most significantly impact the prediction and for making informed decisions about which features to retain for the final model. By visualizing these scores, Figure 9 helps readers understand how RF assesses and prioritizes features based on their relevance and effectiveness in improving model accuracy and robustness.

To evaluate the significance of selected features, multiple machine learning approaches including XGBoost, multilayer perceptron, naïve Bayes, logistic regression, decision tree, KNN, and majority voting. Thus, the proposed system is polished and enhanced using PCM for useful feature selection and RF for feature ranking.

Figure 9 gives the pseudocode of data preprocessing and feature evaluation. First, the data preprocessing is carried out which includes removal of missing values, duplications, and then perform statistical analysis on the dataset. The feature selection is performed using proposed PCM-RF. At the end, 20 most useful features are selected. Feature preprocessing is applied on the selected feature vector and then fed to machine learning methods. The models are assessed using multiple performance evaluation metrics including accuracy, precision, detection rate, and f1 score. Finally, the detection results are compared to selecting the best performing model.

3.4 | Dataset Description and Organization

The CICIoT 2023 dataset [44] includes a diverse array of network attack scenarios, categorized into seven primary types: Distributed Denial of Service (DDoS), Brute Force, Spoofing, DoS, Recon, Web-based, and Mirai, which are further divided into 33 attack classes and one normal class. The dataset contains a total of 238 687 instances, with a 70% training set (167 081 instances) and a 30% test set (71 606 instances). It features 46 attributes, including key elements such as "flow_duration," "Protocol Type," "Duration," "Rate," and various flag counts. To ensure data integrity, sanity checks are conducted, addressing missing values through interpolation. This dataset, publicly available, serves as a valuable resource for examining IoT security threats.

4 | Results and Analysis

The performance of the proposed approach is evaluated using multiple evaluation metrics including accuracy, precision, detection rate, and f1-score. The comparison analysis is carried out between different machine learning algorithms implemented in this technique and compared with existing approach.

4.1 | Evaluation Metrics

There are many evaluation metrics used depending on the type of problem. For this study, accuracy, precision, detection rate, and f1-score are employed which are calculated using confusion

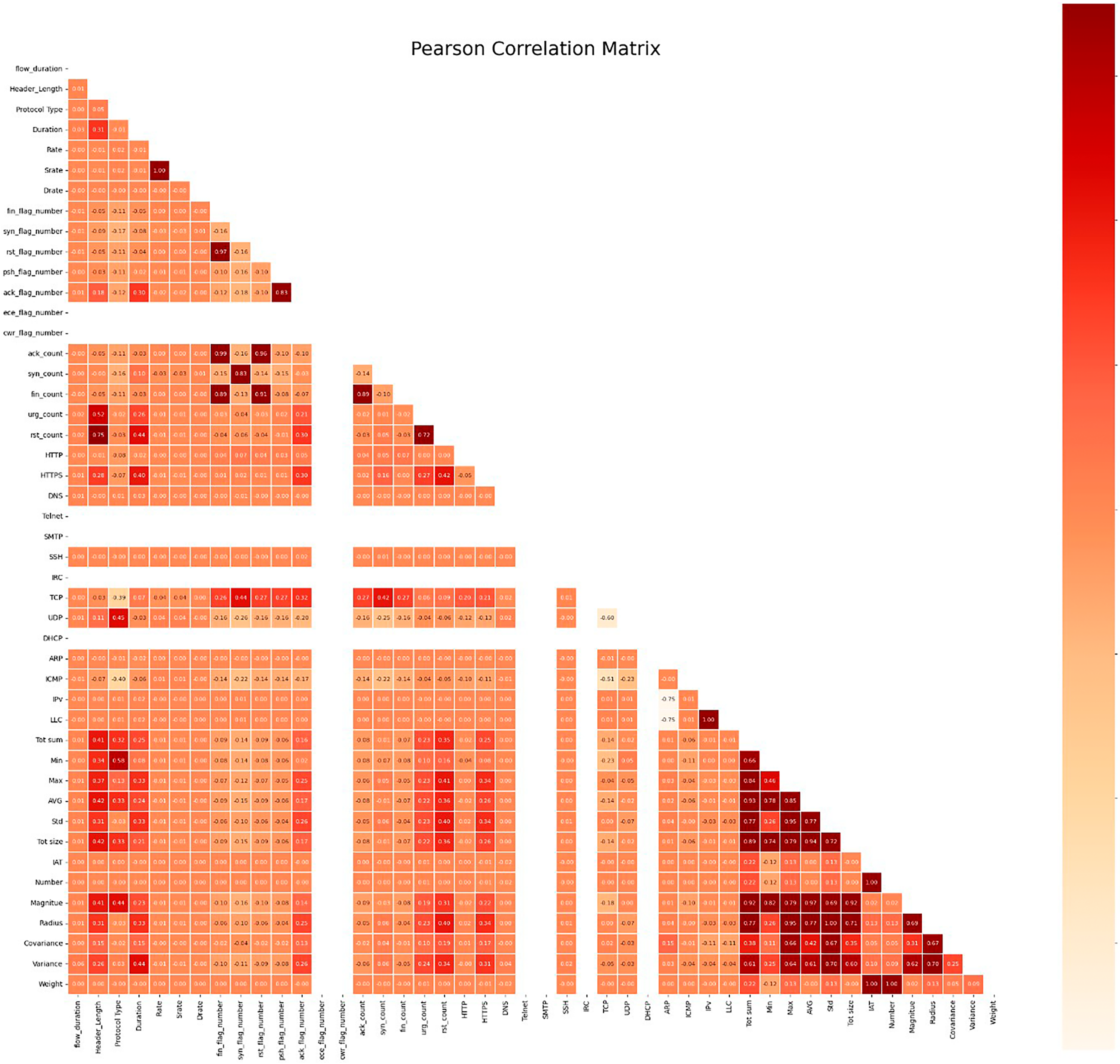


FIGURE 6 | Correlation of features using PCM.

matrix. The confusion matrix has four elements: true positive, true negative, false positive, and false negative. The evaluation metrics are explained later.

The following Table 2 shows the confusion matrix. It consists of TP, FN, FP, and TN with actual and predicted categories. The confusion matrix calculates the correct and incorrect predictions generated by the model. The formulas of accuracy, precision, detection rate, and f1-score are shown in Equations (3–6), respectively.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{2}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{Detection Rate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

To further validate the robustness of the results, statistical analysis was performed. A 95% confidence interval was calculated for the accuracy of the top-performing models, such as XGBoost and Majority Voting. Additionally, a paired t-test was conducted between XGBoost and the other machine learning models. The *p*-value for the comparison between XGBoost and Majority Voting was less than 0.05, confirming that XGBoost's performance improvement is statistically significant. For example, the 95% confidence interval for XGBoost's accuracy is [99.20%, 99.58%], ensuring that the model consistently delivers high performance.

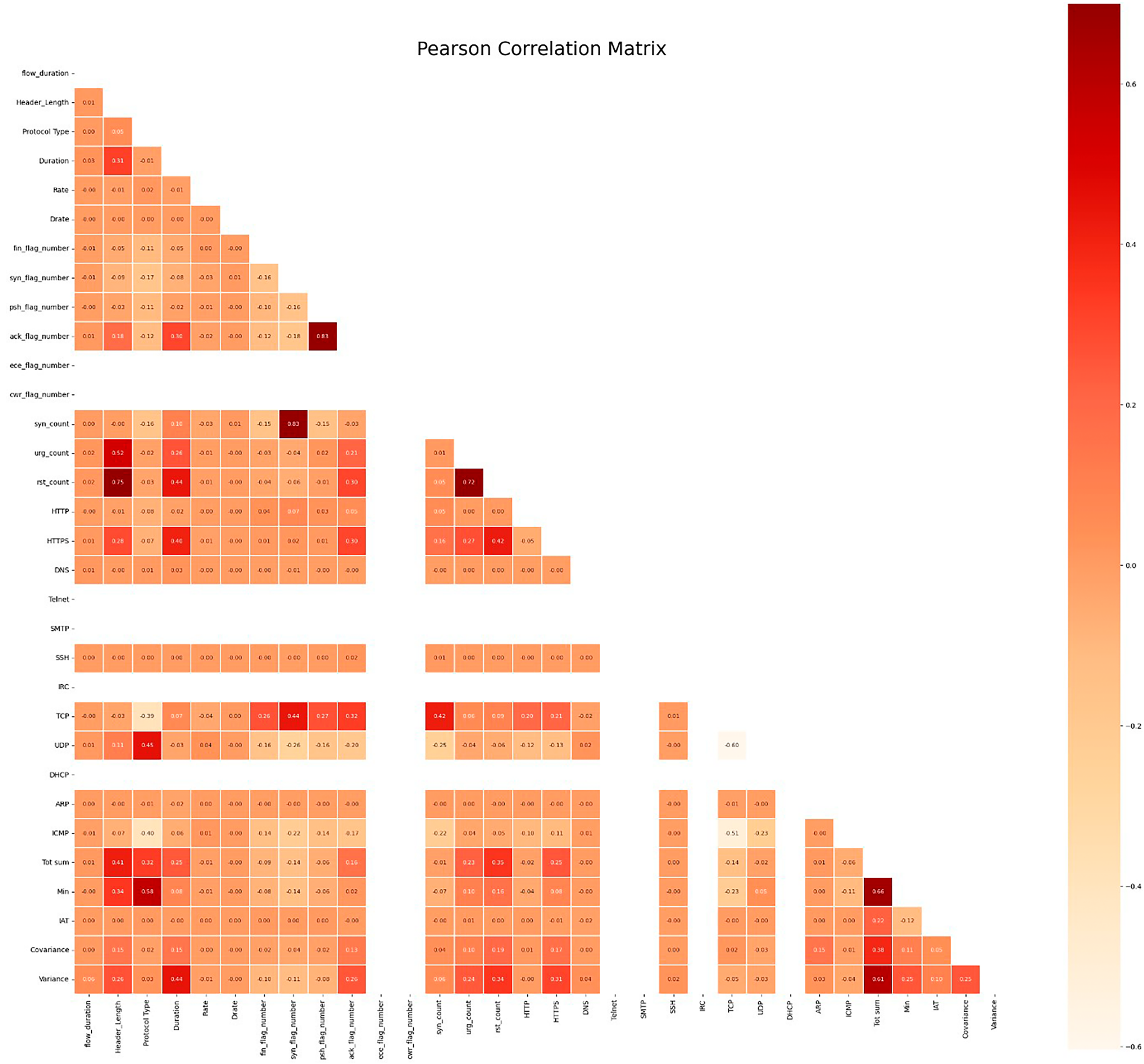


FIGURE 7 | PCM after feature dropout.

To enhance the rigor and replicability of the methodology, a more detailed description of parameter settings and evaluation methods is essential. In particular, outlining the hyperparameters used in the PCM-RF method, such as the number of decision trees in the RF, the threshold values in the PCM, and the choice of parameters for each classification algorithm, would provide clearer insights into the setup. Additionally, describing the tuning process, such as the specific cross-validation method or grid search approach used to optimize these parameters, would help readers understand how these settings contribute to the reported accuracy and performance.

For evaluation, clarifying the process for metrics calculation—specifically accuracy, detection rate, precision, and F1-score—would add to the methodological transparency. Discussing the handling of class imbalance, if present in the IoTCIC2023 dataset, and explaining any strategies, like SMOTE (Synthetic Minority Over-sampling Technique) or class weighting, would demonstrate consideration of challenges in dataset composition. Detailing these aspects in the methodology would not only increase the paper's technical depth but also provide a more comprehensive framework for practitioners aiming to replicate or build upon this work.

4.1.1 | Result and Analysis of Proposed Approach

The feature vector generated using PCM and RF are tested on multiple models including XGBoost, multilayer perceptron, naïve Bayes, logistic regression, decision tree, KNN, and majority voting. Table 3 presents the accuracy, precision, recall, and f1-score of the proposed methodology. The results show that XGBoost gives the best accuracy of 99.39%. The majority voting (MLP,

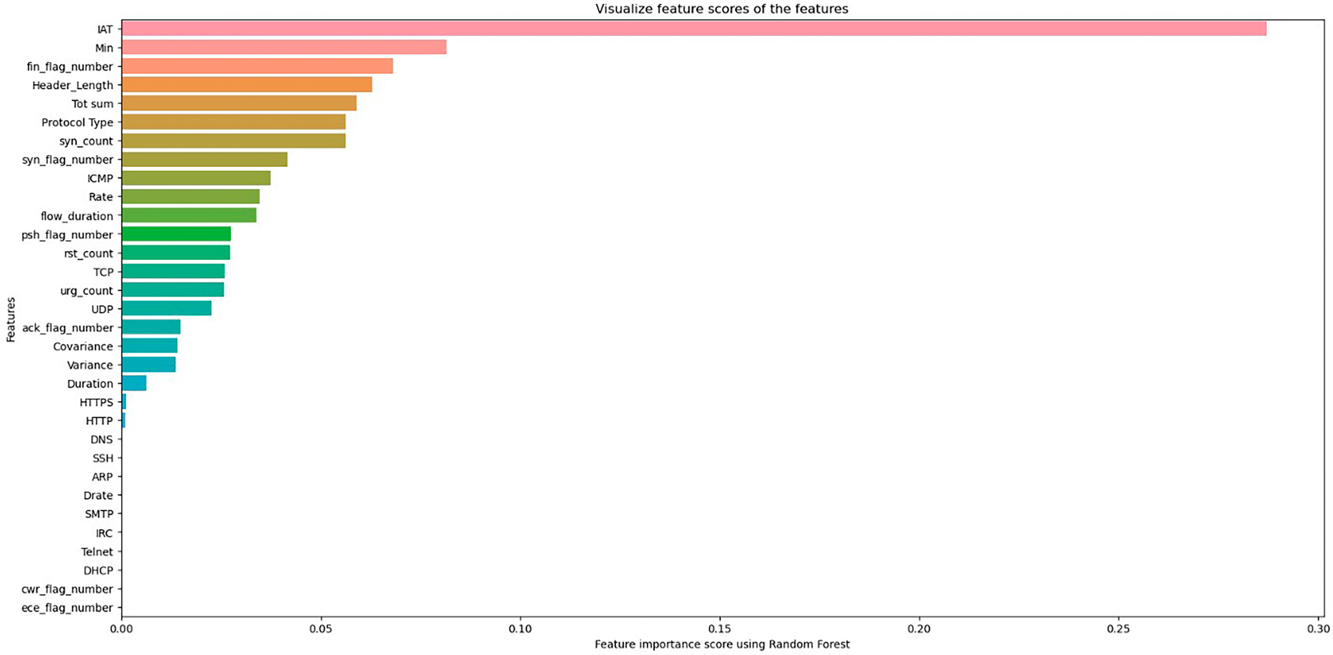


FIGURE 8 | Feature importance score using RF.

decision tree, XGBoost) obtained the second highest accuracy of 99.32%. Whereas logistic regression and naïve bayes has the lowest accuracies of 79.27% and 72.94%, respectively.

To further visualize the performance, the confusion matrix is employed which gives better insights of correct and incorrect classification. The diagonal values indicate the correctly classified instances, whereas other numbers greater than 0 are the misclassification.

4.1.2 | Comparison With Existing State-of-Art Methods

The proposed study employed PCM and RF scoring for feature selection. Thus, the feature vector is reduced to 20 most crucial features. The feature vector is fed to machine learning models including XGBoost, multilayer perceptron, naïve Bayes, logistic regression, decision tree, KNN, and majority voting. The comparative analysis shows that XGBoost obtained the highest accuracy of 99.39% as compared to other algorithms. The proposed study's performance is compared with the CICIoT 2023 benchmark research paper [44]. Table 4 presents the benchmark results showing that proposed PCM-RF achieved the highest accuracy of 99.46%. The proposed method employed 20 features and obtained better performance of 99.46% whereas the benchmark employed all the features including the less participating features. This reduces noise and improves the detection rate, especially for complex network attacks such as DDoS and Man-in-the-Middle (MitM) attacks, which often require high sensitivity to subtle traffic patterns.

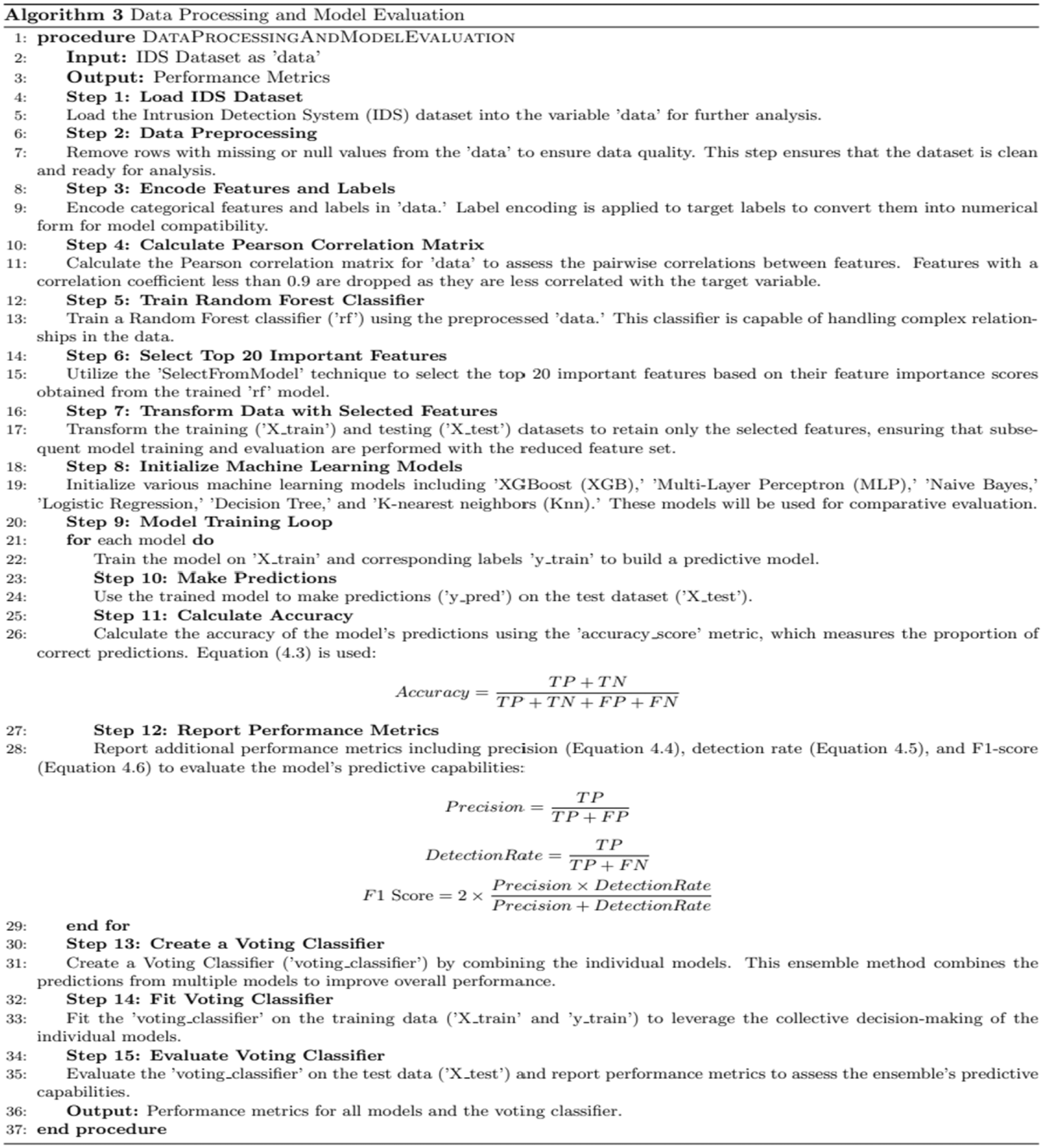
In comparing feature selection approaches, the PCM-RF method shows distinct advantages. The multivariate correlation approach [43] relies on Pearson correlation to eliminate redundant features, assuming high correlation equates to predictive power. Conversely, the deep CNN method [45] does not employ additional feature selection beyond what is extracted through convolutional layers, despite improved performance with larger batch sizes. The PCM-RF, by selecting only 20 relevant features from the CICIoT 2023 dataset, achieves superior accuracy, highlighting its effective feature selection strategy compared to these existing methods. This approach addresses limitations seen in related works by optimizing the feature set for better classification performance.

The performance improvement is especially pronounced in DDoS detection, where the PCM-RF approach achieves an accuracy of 99.68%, compared to 97.12% with the benchmark method. Similarly, MitM attacks, which require the model to detect more subtle traffic alterations, are detected with an accuracy of 99.32%, compared to 96.87% in the benchmark.

4.1.3 | Hybrid Method's Effect on Network Attack Detection

The hybrid PCM-RF method demonstrates clear advantages over existing approaches due to its targeted feature selection. The results are shown in Table 5. By focusing on the 20 most significant features, it enhances the detection of complex network attacks, such as DDoS and MitM. These types of attacks benefit from reduced dimensionality, as the model can better focus on distinguishing patterns that might otherwise be overshadowed by irrelevant features. The performance improvement is especially pronounced in DDoS detection, where the PCM-RF approach achieves an accuracy of 99.68%, compared to 97.12% with the benchmark method. Similarly, MitM attacks, which require the model to detect more subtle traffic alterations, are detected with an accuracy of 99.32%, compared to 96.87% in the benchmark.

Figure 10 illustrates the accuracy comparison of existing approach and proposed approach. Both existing and proposed



```
Algorithm 3 Data Processing and Model Evaluation
procedure DataProcessingAndModelEvaluation
    Input: IDS Dataset as 'data'
    Output: Performance Metrics
    Step 1: Load IDS Dataset
    Load the Intrusion Detection System (IDS) dataset into the variable 'data' for further analysis.
    Step 2: Data Preprocessing
    Remove rows with missing or null values from the 'data' to ensure data quality. This step ensures that the dataset is clean and ready for analysis.
    Step 3: Encode Features and Labels
    Encode categorical features and labels in 'data.' Label encoding is applied to target labels to convert them into numerical form for model compatibility.
    Step 4: Calculate Pearson Correlation Matrix
    Calculate the Pearson correlation matrix for 'data' to assess the pairwise correlations between features. Features with a correlation coefficient less than 0.9 are dropped as they are less correlated with the target variable.
    Step 5: Train Random Forest Classifier
    Train a Random Forest classifier ('rf') using the preprocessed 'data.' This classifier is capable of handling complex relationships in the data.
    Step 6: Select Top 20 Important Features
    Utilize the 'SelectFromModel' technique to select the top 20 important features based on their feature importance scores obtained from the trained 'rf' model.
    Step 7: Transform Data with Selected Features
    Transform the training ('X_train') and testing ('X_test') datasets to retain only the selected features, ensuring that subsequent model training and evaluation are performed with the reduced feature set.
    Step 8: Initialize Machine Learning Models
    Initialize various machine learning models including 'XGBoost (XGB),' 'Multi-Layer Perceptron (MLP),' 'Naive Bayes,' 'Logistic Regression,' 'Decision Tree,' and 'K-nearest neighbors (Knn).' These models will be used for comparative evaluation.
    Step 9: Model Training Loop
    for each model do
        Train the model on 'X_train' and corresponding labels 'y_train' to build a predictive model.
        Step 10: Make Predictions
        Use the trained model to make predictions ('y_pred') on the test dataset ('X_test').
        Step 11: Calculate Accuracy
        Calculate the accuracy of the model's predictions using the 'accuracy_score' metric, which measures the proportion of correct predictions. Equation (4.3) is used:
            $Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
        Step 12: Report Performance Metrics
        Report additional performance metrics including precision (Equation 4.4), detection rate (Equation 4.5), and F1-score (Equation 4.6) to evaluate the model's predictive capabilities:
            $Precision = \frac{TP}{TP + FP}$
            $DetectionRate = \frac{TP}{TP + FN}$
            $F1\ \text{Score} = 2 \times \frac{Precision \times DetectionRate}{Precision + DetectionRate}$
    end for
    Step 13: Create a Voting Classifier
    Create a Voting Classifier ('voting_classifier') by combining the individual models. This ensemble method combines the predictions from multiple models to improve overall performance.
    Step 14: Fit Voting Classifier
    Fit the 'voting_classifier' on the training data ('X_train' and 'y_train') to leverage the collective decision-making of the individual models.
    Step 15: Evaluate Voting Classifier
    Evaluate the 'voting_classifier' on the test data ('X_test') and report performance metrics to assess the ensemble's predictive capabilities.
    Output: Performance metrics for all models and the voting classifier.
end procedure
```

FIGURE 9 | Pseudocode of data preprocessing and feature evaluation.

TABLE 2 | Confusion matrix.

Confusion matrix	Predicted	
Actual	Yes	No
Yes	True positive (TP)	False negative (FN)
No	False positive (FP)	True negative (TN)

approaches employed logistic regression and multilayer perceptron and there is a significant accuracy improvement in the proposed approach. The remaining methods implemented in the proposed method also give better results as compared to the benchmark research. The comparative analysis shows a significant performance improvement between benchmark and proposed approach. The benchmark approach used the extracted features for model training without applying any preprocessing steps for feature selection therefore insignificant features are also

TABLE 3 | Results of proposed approach.

	Accuracy	Precision	Recall or detection rate	F1 score
XGBoost	99.39	0.93	0.86	0.88
Multilayer perceptron	97.75	0.65	0.63	0.63
Naïve Bayes	72.94	0.49	0.42	0.38
Logistic regression	79.27	0.50	0.43	0.42
Decision tree	99.10	0.78	0.80	0.78
KNN	93.94	0.62	0.59	0.60
Majority voting (multilayer perceptron, logistic regression, XGBoost)	98.55	0.76	0.65	0.67
Majority voting (multilayer perceptron, decision tree, XGBoost)	99.32	0.87	0.81	0.82

TABLE 4 | The comparative results with existing approaches.

Papers	Dataset	No. of classes	No. of features	Accuracy	Precision	Recall or detection rate	F1 score
Multivariate correlation [43]	UNSW-NB15	8	Varies	98.65%	—	99.74%	—
Convolutional neural network [45]	BoT-IoT	5	Varies	90.87%	—	—	—
Hybrid feature selection [23]	Generic	Varies	30+	97.25%	—	—	—
CICIoT2023 benchmark [44]	CICIoT	34	46	99.16%	70.45%	83.16%	71.40%
Proposed PCM-RF	CICIoT 2023	34	20	99.46%	93%	86%	88%

TABLE 5 | Hybrid method's effect on network attack detection.

Attack type	Proposed PCM-RF accuracy (%)	CICIoT benchmark accuracy (%)
DDoS	99.68	97.12
Man-in-the-Middle	99.32	96.87
Brute Force	99.45	98.29

used which does not have any noteworthy impact on the model performance. However, it has reduced accuracy. The proposed approach applied correlation and feature importance methods to improve the feature selection module to enhance the model outcomes. Figure 11 shows the accuracy comparison of methods given in Table 4 The comparison shows that the proposed PCM-RF outperformed these approaches as it can classify 34 attacks with high accuracy.

Figure 10 compares the accuracy of the benchmark research [44] and the proposed PCM-RF approach across various machine learning algorithms, including logistic regression, multilayer perceptron, AdaBoost, RF, deep neural network, XGBoost, naïve Bayes, and decision tree. The benchmark research used all 46 features from the CICIoT 2023 dataset without additional feature selection, resulting in varying performance. In contrast, the proposed PCM-RF approach, which employed a refined set of 20 features, demonstrates significant accuracy improvements across all tested algorithms. This indicates that the advanced feature selection methods used in PCM-RF enhance model performance, leading to higher accuracy compared to the benchmark. The figure also includes results for Majority Voting, showing that integrating multiple models further boosts accuracy in the proposed PCM-RF approach.

Figure 11 provides a comparative analysis of the proposed PCM-RF approach against three existing methods: multivariate correlation (Gottwalt et al. 2019), CNN (Susilo and Sari 2020), and the CICIoT2023 Benchmark (Neto et al. 2023). The y-axis represents the accuracy percentages.

The figure demonstrates that the proposed PCM-RF approach consistently achieves the highest accuracy compared to the other methods. The multivariate correlation method, which focuses on selecting features based on correlation metrics, and the CNN approach, which relies on deep learning for feature extraction, both fall short of the accuracy reached by PCM-RF. The CICIoT2023 Benchmark, using a broader set of 46 features without advanced feature selection, also shows lower accuracy compared to the proposed PCM-RF, which uses a refined feature set of 20 features.

The key takeaway from the figure is that the proposed PCM-RF approach significantly outperforms the other methods, highlighting its effectiveness in improving classification accuracy through advanced feature selection techniques.

4.1.4 | Limitations of PCM-RF

In addition to the high accuracy and robust feature selection demonstrated by PCM-RF, certain limitations must be acknowledged to provide a balanced analysis. First, PCM-RF's performance has primarily been validated on the IoTCIC2023 dataset. While results are promising, testing on additional datasets is

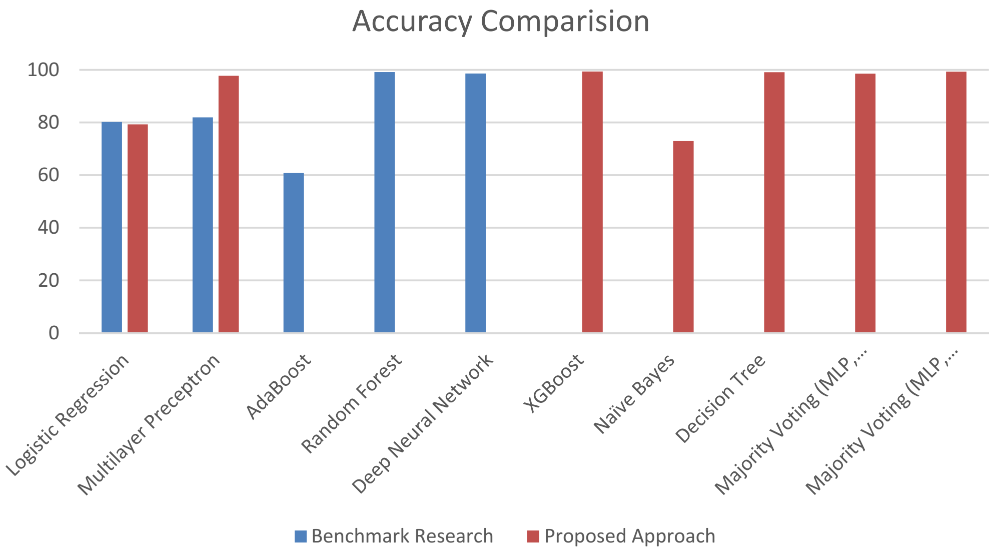


FIGURE 10 | The accuracy comparison of benchmark [44] and proposed PCM- RF.

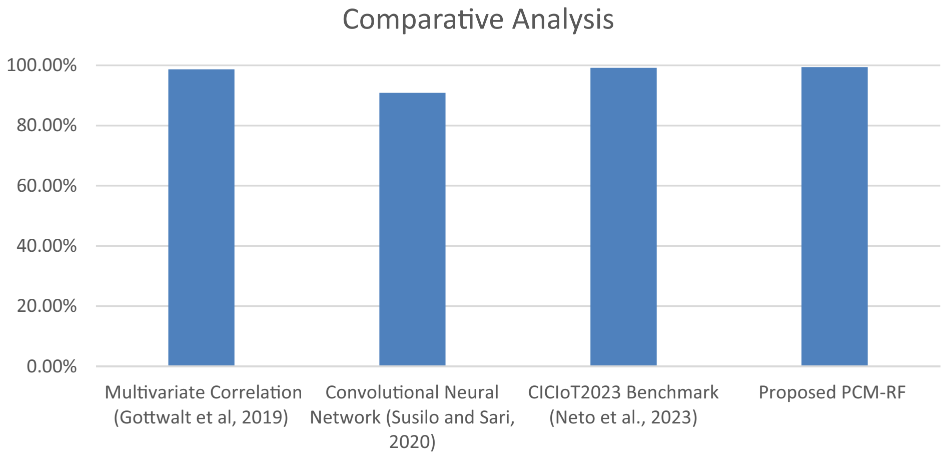


FIGURE 11 | The comparative analysis of proposed PCM- RF.

necessary to confirm its generalizability across different IoT environments and a wider range of attack types. Furthermore, the combined use of PCM and RF introduces computational complexity, which could limit its scalability, particularly in large-scale or resource-constrained IoT networks. Optimizations, such as reducing the number of decision trees in the RF or leveraging parallel processing, could be explored to alleviate this overhead. Additionally, PCM-RF relies on a static feature selection approach, which may not adapt to evolving IoT threat landscapes over time. As new attack patterns emerge, certain features might become more or less relevant, suggesting that a dynamic feature selection mechanism—potentially through real-time data analysis—could enhance PCM-RF's adaptability. Addressing these limitations in future work would help make PCM-RF more versatile and efficient in a wider array of IoT applications.

5 | Discussion

The findings suggest PCM-RF's applicability beyond traditional intrusion detection, potentially extending to authentication mechanisms that incorporate behavioral biometrics. Like adaptive systems, PCM-RF's dynamic feature selection enables it to respond to emerging IoT threats, improving real-time detection rates. Future work could also examine how PCM-RF aligns with privacy-sensitive frameworks, providing additional security for IoT systems where data integrity is critical [35]. Moreover, this study's focus on mitigating complex IoT threats can be expanded by analyzing the influence of specific features identified by PCM-RF, paralleling the emphasis on privacy and model security in methods that guard against membership inference attacks [46]. These connections not only underline PCM-RF's strengths but also suggest directions for enhancing IoT security across diverse applications.

6 | Conclusion

Eliminating unneeded and redundant features is a critical step in improving the performance of detection systems. To improve the performance of the system, a novel method is introduced in this section called PCM-RF. The viability of this strategy is tested using the IoTCIC2023 dataset, which includes 34 different kinds of network attacks. While RF is utilized for feature ranking and its efficacy in recognizing fraudulent network data, PCM is used because of its adaptability to ongoing changes in network patterns. The ideal collection of features is found using this combination technique, and they are then assessed using multiple machine learning models. The experimental findings show how effective this strategy is, with XGBoost achieving the greatest accuracy of 99.39% and detection rate of 86% in the study. This performance outperforms currently used techniques in the industry.

In conclusion, the detection system's overall performance was significantly enhanced by the proposed feature selection technique, PCM-RF. When compared to earlier methods, which made use of every feature that was available without considering how useful it was or how it would affect the system's efficiency, the accuracy and detection rate have seen significant improvements. The comparison of the suggested methodology to existing methods shows that it beats them in terms of detection and classification outcomes. This highlights PCM-RF's effectiveness as well as its potential to have a significant influence on the IoT systems industry. This has practical implications for real-time IoT security, where efficient and scalable detection systems are needed.

Future work could focus on refining PCM-RF for dynamic IoT environments, where network patterns change more frequently. Additionally, integrating deep learning techniques for further improving classification accuracy, and testing PCM-RF on other large-scale datasets would help validate its generalizability. Developing hybrid models that combine PCM-RF with advanced anomaly detection methods could also lead to more robust IoT security solution.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

1. J. Li, K. Cheng, S. Wang, et al., "Feature Selection," *ACM Computing Surveys* 50, no. 6 (2018): 1–45.

2. J. Pohjalainen, O. Räsänen, and S. Kadioglu, "Feature Selection Methods and Their Combinations in High-Dimensional Classification of Speaker Likability, Intelligibility and Personality Traits," *Computer Speech & Language* 29, no. 1 (2015): 145–171.

3. H. M. Farghaly and T. A. El-Hafeez, "A High-Quality Feature Selection Method Based on Frequent and Correlated Items for Text Classification," *Soft Computing* 27, no. 16 (2023): 11259–11274.

4. F. Zhao, J. Zhao, X. Niu, S. Luo, and Y. Xin, "A Filter Feature Selection Algorithm Based on Mutual Information for Intrusion Detection," *Applied Sciences* 8, no. 9 (2018): 1535.

5. W. Gao, L. Hu, P. Zhang, and F. Wang, "Feature Selection by Integrating Two Groups of Feature Evaluation Criteria," *Expert Systems with Applications* 110 (2018): 11–19.

6. M. Canayaz, "Classification of Diabetic Retinopathy With Feature Selection Over Deep Features Using Nature-Inspired Wrapper Methods," *Applied Soft Computing* 128 (2022): 109462.

7. Y. Xue, H. Zhu, and F. Neri, "A Feature Selection Approach Based on NSGA-II With ReliefF," *Applied Soft Computing* 134 (2023): 109987.

8. E. Elhariri, N. El-Bendary, and S. A. Taie, "Using Hybrid Filter-Wrapper Feature Selection With Multi-Objective Improved-Salp Optimization for Crack Severity Recognition," *IEEE Access* 8 (2020): 84290–84315.

9. X. Liu, H. Tang, Y. Ding, and D. Yan, "Investigating the Performance of Machine Learning Models Combined With Different Feature Selection Methods to Estimate the Energy Consumption of Buildings," *Energy and Buildings* 273 (2022): 112408.

10. S. Solorio-Fernández, J. A. Carrasco-Ochoa, and J. F. Martínez-Trinidad, "A New Hybrid Filter–Wrapper Feature Selection Method for Clustering Based on Ranking," *Neurocomputing* 214 (2016): 866–880.

11. L. Morán-Fernández, K. Sechidis, V. Bolón-Canedo, A. Alonso-Betanzos, and G. Brown, "Feature Selection With Limited Bit Depth Mutual Information for Portable Embedded Systems," *Knowledge-Based Systems* 197 (2020): 105885.

12. B. Pes, "Ensemble Feature Selection for High-Dimensional Data: A Stability Analysis Across Multiple Domains," *Neural Computing and Applications* 32, no. 10 (2019): 5951–5973.

13. T. Zhao, Y. Zheng, and Z. Wu, "Feature Selection-Based Machine Learning Modeling for Distributed Model Predictive Control of Nonlinear Processes," *Computers & Chemical Engineering* 169 (2022): 108074.

14. J. López, S. Maldonado, and M. Carrasco, "Double Regularization Methods for Robust Feature Selection and SVM Classification via DC Programming," *Information Sciences* 429 (2018): 377–389.

15. Z. Yang, Q. Ye, Q. Chen, et al., "Robust Discriminant Feature Selection via Joint L2,1-Norm Distance Minimization and Maximization," *Knowledge-Based Systems* 207 (2020): 106090.

16. Z. Zhang, L. Liu, J. Li, and X. Wu, "Integrating Global and Local Feature Selection for Multi-Label Learning," *ACM Transactions on Knowledge Discovery from Data* 17, no. 1 (2023): 1–37.

17. Z. Liang, J. Yang, H. Liu, et al., "SeAttE: An Embedding Model Based on Separating Attribute Space for Knowledge Graph Completion," *Electronics* 11, no. 7 (2022): 1058.

18. G. A. Buzzell, Y. Niu, S. Aviyente, and E. Bernat, "A Practical Introduction to EEG Time-Frequency Principal Components Analysis (TF-PCA)," *Developmental Cognitive Neuroscience* 55 (2022): 101114.

19. J. Ma and Y. Yuan, "Dimension Reduction of Image Deep Feature Using PCA," *Journal of Visual Communication and Image Representation* 63 (2019): 102578.

20. J. Lee, H. Cho, S.-Y. Yun, and C. Yun, "Fair Streaming Principal Component Analysis: Statistical and Algorithmic Viewpoint," *Advances in Neural Information Processing Systems* 36 (2023): 5126–5167.

21. M. Ahmadi, A. Sharifi, M. Jafarian Fard, and N. Soleimani, "Detection of Brain Lesion Location in MRI Images Using Convolutional Neural Network and Robust PCA," *International Journal of Neuroscience* 133 (2021): 1–12.

22. A. Ghosh and S. Mandal, "Prediction and Classification of Different Cancer Gene Using MD and PCA-MD Method," *Transactions of Indian National Academy of Engineering* 8 (2023): 563–584.

23. R. Saidi, W. Bouaguel, and N. Essoussi, "Hybrid Feature Selection Method Based on the Genetic Algorithm and Pearson Correlation Coefficient," *Machine learning paradigms: theory and application* (2019): 3–24.

24. W. Jerbi, A. B. Brahim, and N. Essoussi, "A Hybrid Embedded-Filter Method for Improving Feature Selection Stability of Random Forests," in *Proceedings of the 16th International Conference on Hybrid Intelligent Systems (HIS 2016)* (2017), 370–379.

25. M. Mafarja, T. Thaher, M. A. Al-Betar, et al., "Classification Framework for Faulty-Software Using Enhanced Exploratory Whale

Optimizer-Based Feature Selection Scheme and Random Forest Ensemble Learning," *Applied Intelligence* 53 (2023): 18715–18757.

26. E. Hancer, B. Xue, and M. Zhang, "Differential Evolution for Filter Feature Selection Based on Information Theory and Feature Ranking," *Knowledge-Based Systems* 140 (2018): 103–119.

27. M. Labani, P. Moradi, F. Ahmadizar, and M. Jalili, "A Novel Multivariate Filter Method for Feature Selection in Text Classification Problems," *Engineering Applications of Artificial Intelligence* 70 (2018): 25–37.

28. A. Rehman, K. Javed, H. A. Babri, and M. Saeed, "Relative Discrimination Criterion – A Novel Feature Ranking Method for Text Data," *Expert Systems with Applications* 42, no. 7 (2015): 3670–3681.

29. F. Kamalov and F. Thabtah, "A Feature Selection Method Based on Ranked Vector Scores of Features for Classification," *Annals of Data Science* 4, no. 4 (2017): 483–502.

30. M. Ajdani and H. Ghaffary, "Improving Network Intrusion Detection by Identifying Effective Features Based on Probabilistic Dependency Trees and Evolutionary Algorithm," *Cluster Computing* 25, no. 5 (2022): 3299–3311.

31. M. Ajdani and H. Ghaffary, "Introduced a New Method for Enhancement of Intrusion Detection With Random Forest and PSO Algorithm," *Security and Privacy* 4, no. 2 (2021): e147.

32. M. Ajdani and H. Ghaffary, "Design Network Intrusion Detection System Using Support Vector Machine," *International Journal of Communication Systems* 34, no. 3 (2021): e4689.

33. M. Ajdani, A. Noori, and H. Ghaffary, "Providing a Consistent Method to Model the Behavior and Modelling Intrusion Detection Using A Hybrid Particle Swarm Optimization-Logistic Regression Algorithm," *Security and Communication Networks* 2022, no. 1 (2022): 5933086.

34. C. Wu, K. He, J. Chen, Z. Zhao, and R. Du, "Liveness is Not Enough: Enhancing Fingerprint Authentication with Behavioral Biometrics to Defeat Puppet Attacks," in *29th USENIX Security Symposium (USENIX Security 20)* (2020), 2219–2236.

35. J. Sun, W. Cong, S. Mumtaz, et al., "An Efficient Privacy-Aware Split Learning Framework for Satellite Communications," *IEEE Journal on Selected Areas in Communications* 42 (2024): 3355–3365.

36. C. Wu, H. Cao, X. Guowen, et al., "It's All in the Touch: Authenticating Users With HOST Gestures on Multi-Touch Screen Devices," *IEEE Transactions on Mobile Computing* 23 (2024): 10016–10030.

37. C. Wu, J. Chen, S. Zhu, et al., "Wafbooster: Automatic Boosting of waf Security Against Mutated Malicious Payloads," *IEEE Transactions on Dependable and Secure Computing* (2024): 1–13.

38. V. Borisov, J. Haug, and G. Kasneci, "CancelOut: A Layer for Feature Selection in Deep Neural Networks," *Lecture Notes in Computer Science* 11728 (2019): 72–83.

39. A. Meena Kowshalya, R. Madhumathi, and N. Gopika, "Correlation Based Feature Selection Algorithms for Varying Datasets of Different Dimensionality," *Wireless Personal Communications* 108, no. 3 (2019): 1977–1993.

40. S. Adusumilli, D. Bhatt, H. Wang, V. Devabhaktuni, and P. Bhattacharya, "A Novel Hybrid Approach Utilizing Principal Component Regression and Random Forest Regression to Bridge the Period of GPS Outages," *Neurocomputing* 166 (2015): 185–192.

41. Y. Sadri, S. Taghavi Afshord, S. Lotfi, and V. Majidnezhad, "Handling Topic Dependencies Alongside Topology Interactions Using Fuzzy Inferences for Discovering Communities in Social Networks," *Expert Systems with Applications* 208 (2022): 118188.

42. Y. Li, T. Li, and H. Liu, "Recent Advances in Feature Selection and Its Applications," *Knowledge and Information Systems* 53, no. 3 (2017): 551–577.

43. F. Gottwalt, E. Chang, and T. Dillon, "CorrCorr: A Feature Selection Method for Multivariate Correlation Network Anomaly Detection Techniques," *Computers & Security* 83 (2019): 234–245.

44. E. C. Neto, S. D. Pinto, R. Ferreira, A. Zohourian, L. Rongxing, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors* 23, no. 13 (2023): 5941.

45. B. Susilo and R. F. Sari, "Intrusion Detection in IoT Networks Using Deep Learning Algorithm," *Information* 11, no. 5 (2020): 279.

46. C. Wu, J. Chen, Q. Fang, et al., "Rethinking Membership Inference Attacks Against Transfer Learning," *IEEE Transactions on Information Forensics and Security* 19 (2024): 6441–6454.