IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

An Enhanced Machine Learning-Based Intrusion Detection System for IoT Networks Using Feature Engineering and Data Preprocessing

Vaishnavi Nandu Shelke, Prof. J. K. Shimpi, Prof. S. D. Satre, Dr. H. B. Jadhav
Department of Computer Engineering
Adsul's Technical Campus, Maharashtra, India.

Abstract : *The rapid expansion of Internet of Things (IoT) devices has created new opportunities for smart and connected applications, but it has also increased the number of potential security vulnerabilities. IoT networks generate large volumes of heterogeneous traffic, making the detection of malicious activities a challenging task. Intrusion Detection Systems (IDS) based on machine learning can identify abnormal network behaviour and provide an effective security mechanism for IoT environments. However, the performance of a machine learning model strongly depends on the quality of the input data, preprocessing techniques, feature representation, and class distribution.*

This research focuses on developing a machine learning-based IoT intrusion detection system with particular emphasis on data preprocessing and feature generation. The proposed framework includes data cleaning, handling of missing values, encoding of categorical attributes, feature reduction, and feature generation before applying a machine learning classifier. The ToN-IoT dataset is considered for evaluating the proposed approach because it contains normal and malicious IoT network traffic representing different attack categories. Random Forest is considered as the primary classification algorithm because of its ability to handle high-dimensional data, nonlinear relationships, and multiple classes while maintaining relatively low computational complexity. Appropriate techniques for handling class imbalance are also incorporated to improve the detection of minority attack categories.

The main objective of this study is to investigate whether a carefully designed preprocessing and feature engineering pipeline can improve the performance of a classical machine learning model for IoT intrusion detection. The study also considers important evaluation measures such as accuracy, precision, recall, F1-score, and ROC-AUC to provide a comprehensive assessment of detection performance. The proposed approach aims to provide an efficient, interpretable, and practical alternative to highly complex deep learning-based intrusion detection systems, particularly for resource-constrained IoT environments.

Keywords: Internet of Things, Intrusion Detection System, Machine Learning, Random Forest, Feature Engineering, Data Preprocessing, ToN-IoT, Cybersecurity, IoT Security, Anomaly Detection.

I. INTRODUCTION

The Internet of Things (IoT) has become an important part of modern computing by connecting a large number of physical devices, sensors, applications, and network services. IoT technology is widely used in smart homes, healthcare systems, industrial automation, transportation, energy management, and other applications. The increasing number of connected devices generates a large volume of network traffic and creates a broader environment for potential cyberattacks. Therefore, protecting IoT networks from unauthorized activities and malicious traffic has become an important cybersecurity requirement.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

IoT environments are particularly challenging to secure because they contain heterogeneous devices with different hardware capabilities, communication protocols, and security requirements. Many IoT devices have limited processing power, memory, and energy resources. These limitations make it difficult to deploy security solutions that require high computational resources. In addition, the continuously changing nature of IoT traffic makes traditional security mechanisms less effective against previously unknown or evolving attacks.

Intrusion Detection Systems (IDS) are commonly used to monitor network traffic and identify suspicious or malicious activities. Traditional IDS techniques generally depend on predefined signatures or manually designed rules. Although signature-based systems can effectively detect known attacks, they may have difficulty identifying new or modified attack patterns. Machine Learning (ML) provides an alternative approach by learning patterns from network traffic and using these patterns to classify normal and malicious activities.

Machine learning-based IDS solutions have gained considerable attention because they can automatically identify relationships and patterns within large network datasets. Algorithms such as Random Forest, Support Vector Machine, Decision Tree, Logistic Regression, and other ensemble methods have been applied to intrusion detection. At the same time, Deep Learning (DL) techniques such as Convolutional Neural Networks and Long Short-Term Memory networks have also been investigated because of their ability to learn complex patterns from large datasets.

Although deep learning can provide strong detection capabilities, its higher computational requirements, training complexity, and comparatively limited interpretability can create challenges for practical IoT deployment. Classical machine learning models can provide a simpler and more computationally efficient alternative, particularly when the input data is properly prepared. Therefore, the quality of data preprocessing and feature engineering becomes an important factor in achieving reliable intrusion detection performance.

Data preprocessing involves several operations such as handling missing values, converting categorical information into numerical representations, removing irrelevant attributes, and reducing redundant or highly correlated features. Feature engineering can further improve the representation of network traffic by generating or selecting informative attributes that help the classifier distinguish between normal and malicious activities.

In this study, a machine learning-based IoT intrusion detection framework is investigated using the ToN-IoT dataset. The framework emphasizes structured preprocessing, feature engineering, class imbalance handling, and machine learning-based classification. Random Forest is considered as the primary classification technique because it can handle nonlinear relationships and multiple feature types while providing an efficient classification mechanism.

The overall aim of this research is to investigate the effectiveness of a carefully designed preprocessing and feature generation pipeline for IoT intrusion detection. Instead of depending only on increasingly complex models, the study focuses on improving the quality and representation of the data before classification. This approach can contribute toward developing an intrusion detection system that is accurate, efficient, interpretable, and more suitable for resource-constrained IoT environments.

II. PROBLEM STATEMENT

The rapid growth of IoT networks has increased the amount and diversity of network traffic, creating significant challenges for cybersecurity. IoT devices are often exposed to different types of attacks because of their heterogeneous architecture, limited computational resources, and continuously changing communication patterns. Traditional intrusion detection techniques may not be sufficient to identify evolving and previously unseen malicious activities.

Machine learning-based intrusion detection systems provide an effective approach for identifying abnormal network behaviour. However, the performance of these systems depends strongly on the quality of the input data. IoT network datasets may contain missing values, categorical attributes, redundant features, highly correlated variables, and imbalanced attack classes. If these issues are not properly addressed, the classification model may produce biased or unreliable predictions, particularly for less frequent attack categories.

Existing research has increasingly focused on developing complex deep learning architectures for IoT intrusion detection. Although such models can learn complex patterns, they may require greater computational resources and can

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

be difficult to interpret. Classical machine learning algorithms can provide a simpler alternative, but their effectiveness may be reduced when they are applied directly to poorly processed data.

Therefore, the problem addressed in this research is to develop an effective IoT intrusion detection approach that improves the quality of network data through appropriate preprocessing and feature engineering before applying a classical machine learning algorithm. The study aims to investigate whether a properly prepared feature set combined with Random Forest classification can provide reliable detection of different types of IoT network attacks while maintaining lower model complexity and better interpretability.

The research specifically focuses on the following challenges:

- Handling missing and inconsistent values in IoT network data.
- Converting categorical network attributes into suitable numerical representations.
- Reducing redundant and highly correlated features.
- Improving the representation of network traffic through feature engineering.
- Addressing class imbalance between normal traffic and different attack categories.
- Developing an efficient machine learning-based classification approach.
- Evaluating the detection performance using appropriate classification metrics.
- Investigating the suitability of classical machine learning for resource-constrained IoT environments.

III. LITERATURE SURVEY

The rapid growth of Internet of Things (IoT) networks has increased the need for efficient and reliable intrusion detection systems (IDSs). Machine learning (ML) techniques have been widely investigated because they can identify malicious traffic based on patterns learned from network data. Recent studies have focused on improving detection performance through feature selection, feature extraction, ensemble learning, and optimized classification methods.

Mohy-Eddine et al. proposed an ensemble learning-based intrusion detection model for Industrial IoT security. Their work demonstrates the effectiveness of combining machine learning algorithms for detecting different types of network attacks [2]. This indicates that ensemble-based approaches can provide robust classification performance in complex IoT environments.

Feature selection and feature extraction are important steps in ML-based IDS development. Alqahtany et al. investigated a Random Forest-based intrusion detection mechanism combined with Enhanced Grey Wolf Optimization for IoT networks [3]. Their approach highlights the importance of selecting informative features before classification. Similarly, Kouassi et al. studied Top-K feature selection using XGBoost, LightGBM, and Random Forest for IoT intrusion detection, showing the relevance of reducing unnecessary features while maintaining useful information [4].

The ToN-IoT dataset has become an important benchmark for evaluating IoT intrusion detection approaches. Dharini et al. investigated intrusion detection on the ToN-IoT dataset using a hybrid feature-selection approach [5]. Their work demonstrates that appropriate feature selection can improve the effectiveness of machine learning models when handling high-dimensional IoT security data.

Eren and Küçük investigated automated feature generation for IoT intrusion detection using the ToN-IoT dataset [6]. Their study is particularly relevant to the present work because it emphasizes the role of automatically generated features in improving intrusion detection. The same research direction was further developed in their later IEEE Access study, which investigated smart preprocessing and feature generation with a Random Forest classifier [1].

Recent research has also explored more complex feature extraction and hybrid approaches. Silivery et al. proposed a dual-path feature extraction-based hybrid intrusion detection approach for IoT networks [7]. Such approaches demonstrate that the representation of network traffic features can significantly influence IDS performance.

Federated and distributed learning approaches have also been investigated to address privacy, scalability, and distributed IoT environments. Ioannou et al. proposed a machine-learning intrusion detection framework based on

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 4, November 2026 Impact Factor: 8.2

federated learning for Medical IoT security [8]. These approaches are useful for distributed environments, although they can introduce additional computational and communication requirements.

Deep learning has also been extensively applied to IoT intrusion detection. Lazzarini et al. investigated stacking ensembles of deep learning models for IoT intrusion detection [9]. Deep learning can learn complex representations from network traffic, but its higher computational requirements and reduced interpretability can make deployment more challenging in resource-constrained IoT environments.

Based on the reviewed studies, an important research direction is to improve the quality of input data before applying the classification algorithm. Missing values, categorical attributes, redundant features, correlated variables, and class imbalance can negatively affect intrusion detection performance. Therefore, a well-designed preprocessing and feature-engineering pipeline can potentially allow comparatively lightweight machine learning models to achieve strong detection performance.

The present work focuses on this research direction by combining data preprocessing, feature reduction, feature generation, class-imbalance handling, and Random Forest classification using the ToN-IoT dataset. Rather than relying only on a highly complex deep learning architecture, the proposed approach investigates whether careful data preparation and feature engineering can make a classical ML model an effective solution for IoT intrusion detection [1], [5], [6].

IV. EXISTING SYSTEM

Existing IoT Intrusion Detection Systems primarily use signature-based, anomaly-based, machine learning, or deep learning techniques to identify malicious network activities. Traditional signature-based IDS solutions compare observed traffic with predefined attack signatures. They can be effective for known attacks but may have limitations when dealing with new or modified attack patterns.

Machine learning-based IDS approaches address this limitation by learning patterns from previously collected network traffic. Commonly used algorithms include Support Vector Machine, Random Forest, Decision Tree, Logistic Regression, and other classification techniques. These models can classify network traffic into normal and attack categories based on extracted features.

Deep learning-based approaches have also been widely investigated for IoT intrusion detection. Models such as Artificial Neural Networks, Convolutional Neural Networks, and Long Short-Term Memory networks are capable of learning complex patterns from network data. However, these approaches generally involve more complex architectures and can require greater computational resources. The reference paper notes that deep learning models can also have limited interpretability and may be more difficult to deploy and maintain in resource-constrained environments.

A common limitation of existing machine learning-based approaches is that the classification model may receive insufficiently processed data. IoT datasets can contain missing values, categorical attributes, redundant features, correlated variables, and imbalanced attack classes. Applying a classifier directly to such data can affect its ability to correctly identify different attack categories.

Many existing approaches therefore apply basic preprocessing techniques such as categorical encoding and normalization before model training. However, the reference study highlights that greater attention to preprocessing and feature engineering can potentially improve the performance of classical machine learning models.

Limitations of the Existing System

The major limitations of existing IoT intrusion detection approaches are:

1. **High Model Complexity:** Deep learning-based systems may require significant computational resources and complex training procedures.
2. **Limited Interpretability:** Complex deep learning architectures can make it difficult to understand how individual predictions are generated.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

3. **Insufficient Data Preprocessing:** Inadequate handling of missing, categorical, redundant, and correlated features can negatively affect model performance.
4. **Class Imbalance:** Unequal representation of attack categories can cause the model to favor majority classes.
5. **Difficulty in Detecting Minority Attacks:** Rare attack categories may contain fewer training samples, resulting in lower detection performance.
6. **Resource Requirements:** Complex IDS models may not be suitable for IoT environments where devices have limited memory, processing power, and energy.
7. **Feature Representation Issues:** Raw network features may not provide the most informative representation for classification, creating a need for effective feature engineering.

These limitations motivate the development of a simpler machine learning-based approach that gives greater importance to preprocessing, feature engineering, and balanced training before classification.

V. PROPOSED SYSTEM

The proposed system is a machine learning-based IoT Intrusion Detection System designed to improve attack detection by giving greater importance to data preprocessing, feature engineering, and class balancing. Instead of relying only on a complex classification model, the proposed approach focuses on improving the quality of the input data before applying the machine learning algorithm.

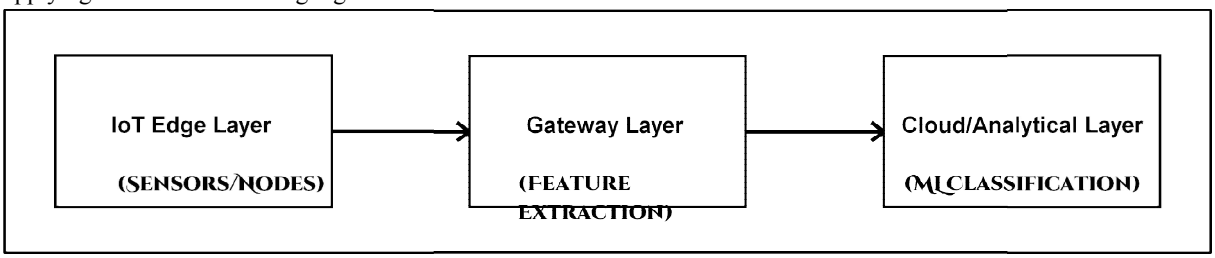


Fig. 1: Proposed IoT Intrusion Detection System Architecture

The proposed framework uses the **ToN-IoT dataset**, which contains network traffic representing normal activities and different categories of cyberattacks. The reference study also uses ToN-IoT for evaluating its IoT intrusion detection framework.

The overall system consists of several sequential stages. First, the input dataset is examined and prepared for processing. Missing or inconsistent values are handled to improve data quality. Categorical attributes are converted into numerical representations so that they can be processed by machine learning algorithms. Irrelevant and highly correlated features are reduced to avoid unnecessary information and improve the efficiency of the classification process.

After preprocessing, feature engineering is performed to obtain a more informative representation of the network traffic. The processed features are then used for handling class imbalance so that minority attack categories receive adequate representation during model training. The reference approach uses a hybrid resampling strategy involving SMOTE and cluster-based under sampling.

The resulting dataset is provided to a **Random Forest classifier** for intrusion detection. Random Forest is selected as the primary machine learning model because it can work with multiple features and nonlinear relationships and can classify different traffic categories using an ensemble of decision trees.

The trained model classifies network traffic into normal or malicious categories. Its performance is evaluated using standard metrics such as **accuracy, precision, recall, F1-score, and ROC-AUC**. These measures provide a broader evaluation of the system instead of relying only on overall accuracy.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

Proposed System Workflow

ToN-IoT Dataset
↓
Data Collection and Inspection
↓
Data Cleaning and Missing Value Handling
↓
Categorical Feature Encoding
↓
Feature Reduction
↓
Feature Engineering / Feature Generation
↓
Class Imbalance Handling
↓
Random Forest Classification
↓
Intrusion Detection
↓
Performance Evaluation

Advantages of the Proposed System

1. **Improved Data Quality:** Preprocessing reduces the effect of missing, inconsistent, and unsuitable data.
2. **Better Feature Representation:** Feature engineering provides more useful information to the classification model.
3. **Balanced Training:** Resampling techniques help improve the representation of minority attack categories.
4. **Lower Complexity:** Random Forest provides an effective classical machine learning solution without requiring a highly complex deep learning architecture.
5. **Interpretability:** Classical machine learning models can be easier to analyse and interpret compared with complex deep learning architectures.
6. **Efficient Intrusion Detection:** The approach aims to provide reliable attack detection while keeping computational requirements comparatively manageable.
7. **Suitable for IoT Environments:** The combination of preprocessing, feature engineering, and classical machine learning is intended to support deployment in environments with limited computational resources.

The proposed system therefore focuses not only on the choice of classifier but also on the complete data preparation pipeline, which is a central aspect of the reference study.

VI. METHODOLOGY

The proposed IoT intrusion detection system follows a sequential methodology in which network traffic data is processed, transformed, balanced, and finally classified using a machine learning model. The methodology focuses on improving the quality of the input data before classification.

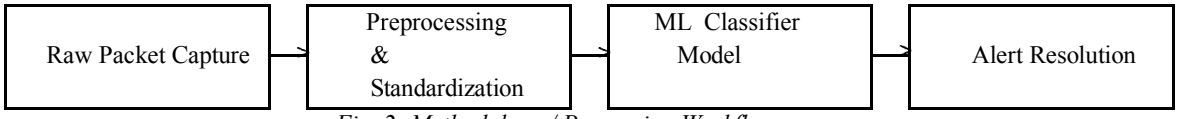


Fig. 2: Methodology / Processing Workflow

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

A. Dataset Selection

The **ToN-IoT dataset** is used as the primary dataset for the proposed IoT intrusion detection system. It is designed for evaluating cybersecurity and intrusion detection techniques in IoT and related network environments. The dataset contains normal network traffic as well as multiple categories of malicious activities.

The dataset provides heterogeneous network-related attributes that can be used to identify differences between legitimate and malicious traffic. Since different attack categories may have different numbers of samples, class imbalance is an important consideration during model development.

B. Data Preprocessing

Data preprocessing is performed before model training to improve the quality and consistency of the dataset. The major preprocessing operations include:

- Identification and handling of missing values.
- Removal of unsuitable or irrelevant attributes.
- Conversion of categorical attributes into numerical form.
- Identification and reduction of highly correlated features.
- Preparation of the resulting data for machine learning.

Proper preprocessing is important because machine learning algorithms depend on the quality and representation of the input features.

C. Categorical Feature Encoding

IoT network datasets may contain categorical attributes that cannot be directly processed by many machine learning algorithms. These attributes are converted into numerical representations using an appropriate encoding technique.

The encoded features are combined with numerical attributes to create a machine-readable representation of the network traffic.

D. Feature Reduction

Network datasets may contain redundant or highly correlated features. Such features can provide similar information and may unnecessarily increase the dimensionality of the dataset.

Feature correlation is therefore analyzed and highly correlated attributes are reduced where appropriate. This step helps create a more compact feature space and can reduce unnecessary computational processing.

E. Feature Generation

After preprocessing, feature engineering is performed to improve the representation of network traffic. New informative features can help the classifier identify relationships that may not be sufficiently represented by the original attributes.

The reference paper uses a feature-generation approach adapted from **Explainable Boosted Linear Regression (EBLR)** and based on iterative tree-based residual analysis.

For the proposed study, feature generation is treated as an important stage of the overall pipeline because the objective is to improve classification through better feature representation rather than relying only on increasing model complexity.

F. Handling Class Imbalance

IoT intrusion datasets can contain unequal numbers of samples for different traffic and attack categories. If the dataset is highly imbalanced, a classifier may become biased toward classes with a larger number of samples.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

To reduce this problem, a hybrid resampling strategy can be applied. The reference study uses **SCUT**, which combines **SMOTE (Synthetic Minority Oversampling Technique)** with cluster-based under sampling to create a more balanced training dataset.

G. Random Forest Classification

The processed and balanced dataset is provided to a **Random Forest classifier**. Random Forest is an ensemble learning algorithm that combines multiple decision trees to perform classification.

For intrusion detection, the classifier learns patterns associated with normal and malicious network traffic from the training data. During testing, the trained model uses the learned patterns to classify previously unseen network traffic.

H. Model Evaluation

The trained model is assessed with standard classification metrics:

Qualities: Evaluates the total number of accurately categorized cases.

Precision: Measures how many instances classified as an attack actually belong to the predicted attack category.

Recall: measures how well the model can spot examples of a certain type of attack.

F1-Score: is a way to show how well a model does by combining both precision and recall.

ROC-AUC measures the model's ability to differentiate between various classes across different classification thresholds**.**

These metrics provide a comprehensive assessment of the intrusion detection performance.

VII. EXPERIMENTAL SETUP

The experimental setup defines the environment, dataset preparation procedure, machine learning model, and evaluation process used to assess the proposed IoT intrusion detection system. The experiment is designed to determine the effectiveness of preprocessing and feature engineering in improving the classification of IoT network traffic.

A. Dataset

The **ToN-IoT dataset** is used for the experimental evaluation. The dataset contains normal and malicious network traffic and includes different categories of cyberattacks. Its heterogeneous nature makes it suitable for evaluating machine learning-based IoT intrusion detection approaches. The reference paper also evaluates its proposed system using the ToN-IoT dataset.

B. Data Preparation

Before training the classifier, the dataset undergoes a sequence of preprocessing operations. Missing values are identified and handled, categorical attributes are converted into numerical representations, and redundant or highly correlated features are reduced.

The processed data is then prepared for feature engineering and class balancing. These steps are performed before model training so that the Random Forest classifier receives a more suitable representation of the network traffic.

C. Training and Testing

The prepared dataset is divided into training and testing data. The training portion is used to learn the relationship between network features and traffic classes, while the testing portion is reserved for evaluating the trained model on unseen samples.

The reference work uses a group-shuffle split together with stratification and applies stratified cross-validation during model development.

D. Classification Model

The main method of classification is through Random Forest. It is an ensemble learning technique that combines multiple decision trees to produce a final prediction. The use of multiple trees helps the model handle complex relationships within the network traffic data.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

The model is trained using the processed feature set and then used to classify network traffic into the corresponding traffic or attack categories.

E. Performance Metrics

The intrusion detection model's performance is measured using these metrics:

Metric	Purpose
Accuracy	Measures the overall percentage of correctly classified samples.
Precision	Measures the correctness of positive attack predictions.
Recall	Measures the ability to identify actual attack samples.
F1-Score	Provides a balance between precision and recall.
ROC-AUC	Measures the ability of the classifier to distinguish between classes.

F. Experimental Objective

The main objective of the experiment is to determine whether appropriate preprocessing, feature engineering, and class balancing can improve the effectiveness of a classical machine learning model for IoT intrusion detection.

The experiment also provides a basis for comparing the proposed classical machine learning approach with more complex intrusion detection approaches reported in existing research.

Important: Actual accuracy, precision, recall, F1-score, and ROC-AUC values should be inserted only after running the implementation. The source paper reports its own experimental values, but those values should not be presented as results of your implementation.

VIII. RESULTS AND DISCUSSION

The performance of the proposed machine learning-based IoT intrusion detection approach was evaluated using the ToN-IoT dataset. The evaluation focused on the effect of data preprocessing and generated features on the performance of the Random Forest classifier. The reported results demonstrate that appropriate preprocessing and feature engineering can significantly improve the performance of a classical machine learning model for IoT intrusion detection.

A. Overall Performance

The model using the original and newly generated features achieved strong overall classification performance. The reported specificity was 99.94%, recall was 99.19%, precision was 99.48%, and F1-score was 99.32%. The model also achieved a ROC-AUC score of 99.99% on both the training and testing datasets.

Evaluation Metric	Reported Performance
Specificity	99.94%
Recall	99.19%
Precision	99.48%
F1-Score	99.32%
ROC-AUC	99.99%

These results indicate that the combination of preprocessing, feature generation, class balancing, and Random Forest classification provides strong discrimination between normal and malicious network traffic. The ROC-AUC value also indicates that the classifier was able to distinguish different classes effectively.

B. Class-wise Performance

The model demonstrated strong performance across several attack categories. For normal traffic, the reported precision, recall, and F1-score were 0.999, 0.988, and 0.994, respectively. For DoS attacks, the corresponding values were 0.985, 0.977, and 0.981. For scanning attacks, precision was 0.984, recall was 0.992, and F1-score was 0.988.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

Attack/Traffic Class	Precision	Recall	F1-Score
Normal	0.999	0.988	0.994
DoS	0.985	0.977	0.981
Scanning	0.984	0.992	0.988
Ransomware	0.989	0.999	0.995
XSS	0.964	0.976	0.970
Password	0.994	0.989	0.992
DDoS	0.970	0.979	0.975
MITM	0.481	0.880	0.620

The results show particularly strong performance for normal traffic, DoS, scanning, ransomware, password, and DDoS attacks. The ransomware class achieved a recall of 0.999, indicating that very few ransomware samples were missed.

C. Effect of Feature Generation

The addition of generated features improved the detection performance for several traffic and attack categories. Improvements were reported for normal traffic, DoS, injection, MITM, scanning, password, ransomware, and XSS classes. The generated features provided additional information that helped the Random Forest classifier distinguish between different network behaviours.

The overall results indicate that feature engineering can be as important as the choice of classification algorithm. Instead of relying only on a complex model architecture, meaningful features extracted from the original network traffic can improve the classification capability of a classical machine learning model.

D. Analysis of MITM Detection

The MITM attack category represented the main limitation of the proposed approach. The reported precision was 0.481, while recall was 0.880 and F1-score was 0.620. The relatively low precision indicates that the classifier generated a comparatively high number of false-positive predictions for this attack category.

This limitation can be associated with the relatively low representation and complexity of MITM attack samples in the dataset. Therefore, additional techniques such as advanced resampling, ensemble learning, synthetic data generation, or hybrid machine learning and deep learning approaches could be investigated to improve MITM detection.

E. Discussion

Overall, the reported findings demonstrate that a carefully designed preprocessing and feature-engineering pipeline can make Random Forest a strong candidate for IoT intrusion detection. The approach provides high classification performance while retaining the advantages of classical machine learning, including comparatively lower computational complexity, easier interpretation, and simpler model maintenance.

The results also demonstrate the importance of handling missing data, categorical attributes, correlated features, and class imbalance before model training. Therefore, effective data preparation should be considered an essential component of an IoT intrusion detection system rather than merely a preliminary processing step.

Note: The numerical values presented in this section are the reported results of the reference study and should not be represented as independently generated experimental results unless the proposed implementation is reproduced and evaluated separately.

IX. CONCLUSION AND FUTURE SCOPE

A. Conclusion

The increasing adoption of Internet of Things (IoT) devices has created a strong need for effective and efficient intrusion detection systems. IoT environments generate large volumes of heterogeneous network data, making the detection of malicious activities a challenging task. This work presents a machine learning-based IoT intrusion

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 4, November 2026 Impact Factor: 8.2

detection approach that focuses on data preprocessing, feature engineering, class imbalance handling, and Random Forest classification.

The proposed approach emphasizes the importance of preparing the input data before applying the classification algorithm. Missing-value handling, categorical feature encoding, feature reduction, feature generation, and class balancing are incorporated into the detection pipeline. These steps are intended to improve the quality of the input data and enable the classifier to distinguish between normal and malicious network traffic more effectively.

Random Forest is selected as the primary classification algorithm because of its ability to handle high-dimensional data and perform classification using multiple decision trees. The proposed methodology is evaluated using standard performance measures such as accuracy, precision, recall, F1-score, and ROC-AUC.

The overall approach demonstrates that effective preprocessing and feature engineering can play an important role in developing a practical machine learning-based IoT intrusion detection system. Rather than depending only on complex deep learning architectures, a carefully designed classical machine learning pipeline can provide an alternative approach for IoT security applications.

B. Future Scope

The proposed work can be extended in several directions:

1. **Evaluation on Additional Datasets:** The system can be tested on other IoT and network intrusion datasets to examine its generalization capability.
2. **Comparison with Other Machine Learning Algorithms:** Algorithms such as Support Vector Machine, Decision Tree, Logistic Regression, XGBoost, and other ensemble methods can be compared with Random Forest.
3. **Advanced Feature Engineering:** Additional feature-generation and feature-selection techniques can be investigated to identify more informative network characteristics.
4. **Real-Time Intrusion Detection:** The proposed methodology can be integrated into a real-time monitoring system for continuous detection of suspicious IoT traffic.
5. **Lightweight Deployment:** The system can be optimized for resource-constrained IoT and edge devices where computational and memory resources are limited.
6. **Improved Minority-Class Detection:** Further work can focus on improving the detection of rare attack categories through advanced sampling and cost-sensitive learning techniques.
7. **Explainable Intrusion Detection:** Explainable AI techniques can be incorporated to provide security analysts with understandable reasons behind the classification decisions.
8. **Hybrid Machine Learning Models:** Future systems can investigate combinations of classical machine learning and deep learning techniques to improve detection performance while maintaining reasonable computational requirements.

REFERENCES

[1] K. K. Eren, K. Küçük, F. Özyurt, and O. H. Alhazmi, "Simple Yet Powerful: Machine Learning-Based IoT Intrusion System With Smart Preprocessing and Feature Generation Rivals Deep Learning," *IEEE Access*, vol. 13, pp. 41435–41455, 2025, doi: 10.1109/ACCESS.2025.3547642.

[2] K. Eren and K. Küçük, "Improving Intrusion Detection Systems for IoT Devices using Automated Feature Generation based on ToN_IoT dataset," in *Proc. 8th Int. Conf. Computer Science and Engineering (UBMK)*, 2023, pp. 276–281.

[3] N. Dharini, V. Janani, and J. Katiravan, "Efficient detection of intrusions in TON-IoT dataset using hybrid feature selection approach," *Scientific Reports*, vol. 16, 2026.

[4] S. S. Alqahtany, A. Shaikh, and A. Alqazzaz, "Enhanced Grey Wolf Optimization (EGWO) and random forest based mechanism for intrusion detection in IoT networks," *Scientific Reports*, vol. 15, 2025.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 4, November 2026
Impact Factor: 8.2

[5] B. M. Kouassi, A. B. Ballo, K. J. Ayikpa, M. Diarra, and M. Z. J. Coulibaly, "Top-K Feature Selection for IoT Intrusion Detection: Contributions of XGBoost, LightGBM, and Random Forest," *Future Internet*, vol. 17, Art. no. 529, 2025.

[6] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrour, and Y. Farhaoui, "An Ensemble Learning Based Intrusion Detection Model for Industrial IoT Security," *Big Data Mining and Analytics*, vol. 6, pp. 273–287, 2023.

[7] A. K. Silivery, K. R. M. Rao, and R. Solleti, "Dual-path feature extraction based hybrid intrusion detection in IoT networks," *Computers & Electrical Engineering*, vol. 122, Art. no. 109949, 2025.

[8] I. I. Ioannou, P. Nagaradjane, P. Angin, P. Balasubramanian, K. J. Kavitha, P. Murugan, and V. Vassiliou, "GEMLIDS-MIOT: A Green Effective Machine Learning Intrusion Detection System based on Federated Learning for Medical IoT network security hardening," *Computer Communications*, vol. 218, pp. 209–239, 2024.