AI SECURITY THREATS IN 5G ENABLED IOT DEVICE MANAGEMENT

1st K Mahendra Kumar, D Bapuji Reddy, P Rushendra
Department of ECE
SRM Institute of Science and Technology
Chennai, India
Email:bd2320, mk1815, rp8813@srmist.edu.in

2nd Dr. S Murugaveni
Assistant Professor, Department of ECE
SRM Institute of Science and Technology
Chennai, India
murugavs@srmist.edu.in

***Abstract*—The integration of 5g with IOT devices enhances the connectivity but introduce some threats at the same time. As 5g connects more devices it sometimes has a weak safety measurement and reduces the privacy and helps in the data leaks. Here comes the AI where it plays a crucial role in managing and securing them. But with the play of AI also have some issues as they can manipulate systems and steal data and some vulnerabilities. To make it strong we need to train different AI models and adapt a different approach for the problems and use past references to tackle a situation and avoid the problems and secure the network or data transfer between multiple devices.**

***Index Terms*—5G, IoT, AI Security, Threat Detection, Blockchain, Privacy**

I. Introduction

The swift development of wireless communication technology has opened the way for common usage of the Internet of Things (IoT), which has made possible seamless communication and data exchange between billions of intelligent devices. Among them, the establishment of fifth-generation (5G) mobile networks is a particularly important milestone. With its promise of ultra-low latency, large-scale connectivity, and improved bandwidth, 5G serves as an enabler of large-scale deployments of IoT in a variety of sectors ranging from smart cities and healthcare to manufacturing and transport.

But this technological development also introduces a new set of cybersecurity issues. The massive number of interlinked devices and the network's heterogeneity make it vulnerable to exploits by malicious users. Problems of unauthorized access, data breaches, Denial-of-Service (DoS) attacks, and man-in-the-middle (MITM) attacks are exacerbated in 5G-IoT networks because of the decentralized and complicated infrastructure.

II. Literature Survey

The integration of Artificial Intelligence (AI) in 5G-enabled Internet of Things (IoT) ecosystems has generated significant research interest due to its potential for enhancing security while also introducing new attack vectors. Several studies have addressed various dimensions of AI, blockchain, and mobile edge computing in this domain. A review of relevant literature is presented below.

A. AI-Driven Threat Detection in IoT

Chris Gilbert et al. [1] discuss the application of AI-based methods in threat detection for IoT devices. The study explores supervised and unsupervised learning models for identifying anomalies and malicious behavior. It also emphasizes the vulnerabilities introduced by AI models themselves, such as adversarial attacks and data poisoning, advocating for continuous retraining and validation.

B. 5G Security Threat Landscape and Blockchain Integration

Mohammad N. Alanazi [2] outlines the security threats in 5G-IoT systems and introduces blockchain-based solutions to mitigate risks. The study proposes a decentralized AI-powered identity management system using self-sovereign identity (SSI) frameworks. This combination enhances authentication and data integrity in distributed environments.

C. Smart City Infrastructure with AI and Drones

Poonam Tiwari et al. [3] investigate drone-based services in 5G smart city networks. They present a secure blockchain-assisted drone traffic management system with AI-based geofencing for real-time surveillance and threat mitigation. Although tailored for drones, the approach is applicable to a broader range of IoT device management applications.

D. Adaptive Machine Learning for IoT Security

Harshit Raheja et al. [4] propose an adaptive machine learning security model for IoT environments. The system dynamically selects optimal algorithms based on the prevailing threat landscape. The work also introduces a zero-trust architecture that mandates continuous authentication across all devices, reducing the risk of lateral attacks.

E. AI for Mobile-Edge Computing Security

Cheng Wang [5] surveys AI-powered techniques for Mobile Edge Computing (MEC). The paper promotes federated learning as a privacy-preserving model training method. It also includes the use of AI for real-time anomaly detection and decentralized threat prevention across edge devices.

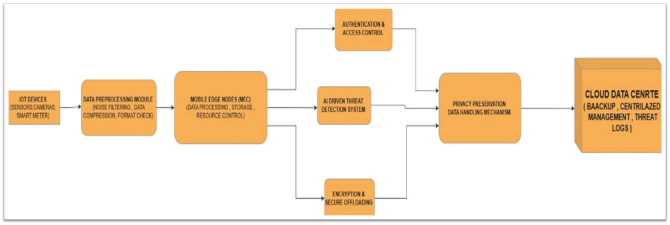

IOT DEVICES
(SENSORS,CAMERAS, SMART METER)
DATA PREPROCESSING MODULE
(NOISE FILTERING , DATA COMPRESSION, FORMAT CHECK)
MOBILE EDGE NODES (MEC)
(DATA PROCESSING , STORAGE , RESOURCE CONTROL)
AUTHENTICATION & ACCESS CONTROL
AI DRIVEN THREAT DETECTION SYSTEM
ENCRYPTION & SECURE OFFLOADING
PRIVACY PRESERVATION DATA HANDLING MECHANISM
CLOUD DATA CENRTE
(BAACKUP , CENTRILAZED MANAGEMENT , THREAT LOGS)

Fig. 1. Block diagram of the proposed 5G-IoT security framework.

F. Hybrid AI-Based Intrusion Detection Systems

Jonghoon [6] focuses on enhancing network security in 5G-based industrial IoT using hybrid AI models. Utilizing the NSL-KDD and CICIDS datasets, the study demonstrates improved accuracy by combining supervised and unsupervised learning. This serves as a strong foundation for lightweight, scalable intrusion detection systems deployable at the edge.

III. Methodology

The proposed security framework for 5G-enabled IoT device management consists of two tightly coupled pipelines: (1) a multi-layer system architecture for data collection, edge preprocessing, secure offloading and access control and (2) an AI-based threat detection workflow with privacy-preserving input handling. We describe each in turn below.

A. System Architecture

Fig.1 presents the overall system model. Raw data from heterogeneous IoT devices (e.g., sensors, cameras, smart meters) is first routed to a *Data Preprocessing Module* where noise filtering, data compression and format checks are applied. The cleaned data is then forwarded to a cluster of *Mobile Edge Nodes (MEC)*, responsible for local data processing, short-term storage and resource orchestration.

Within each MEC node three security services execute in parallel:

- **Authentication & Access Control:** Enforces device identity verification (e.g., blockchain-backed self-sovereign IDs) and fine-grained policies to ensure only authorized entities can exchange data.
- **AI-Driven Threat Detection:** Monitors incoming data streams with machine-learning models to detect anomalies or known attack signatures in real time.
- **Encryption & Secure Offloading:** Encrypts sensitive payloads and offloads non-latency-critical data to the cloud for long-term storage and further analysis.

All outputs converge into a *Privacy Preservation Data Handling Module*, which integrates hashed identifiers and differential privacy techniques before committing logs, alerts and raw backups to a centralized *Cloud Data Centre*.

B. AI-Driven Threat Detection Workflow

The detailed machine-learning pipeline inside the AI-Driven Threat Detection block is illustrated in Fig. 2. This workflow ensures robust classification while preserving device privacy.

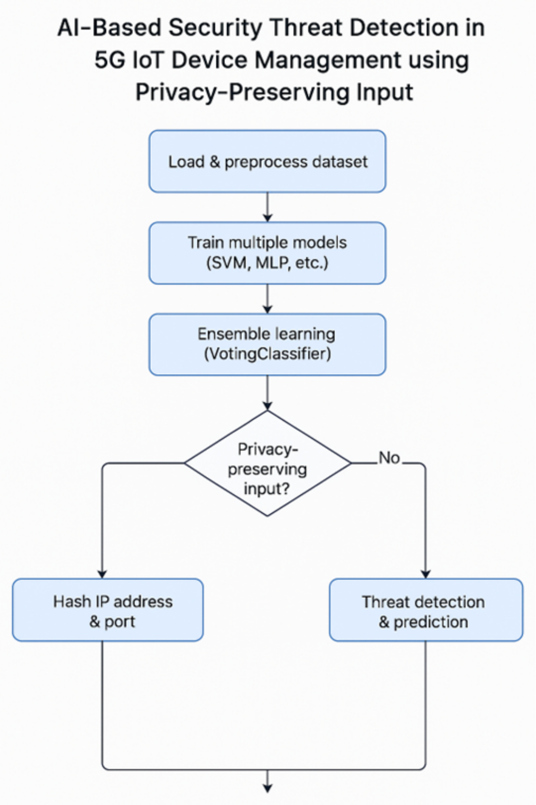

AI-Based Security Threat Detection in 5G IoT Device Management using Privacy-Preserving Input
Load & preprocess dataset
Train multiple models (SVM, MLP, etc.)
Ensemble learning (VotingClassifier)
Privacy-preserving input?
No
Hash IP address & port
Threat detection & prediction

Fig. 2. Flowchart of AI-based threat detection with privacy-preserving input.

a) Data Loading & Preprocessing: Traffic logs, packet headers, and contextual metadata are aggregated from MEC buffers. Feature-engineering routines normalize numeric attributes and encode categorical fields.

b) Base-Learner Training: Multiple classifiers (e.g., Support Vector Machine, Multilayer Perceptron, Random Forest) are trained independently on historical labeled data (e.g., NSL-KDD, CICIDS) to capture diverse attack patterns.

c) Ensemble Learning: A VotingClassifier combines base-learner predictions to improve overall robustness and reduce variance. Both hard and soft voting strategies are evaluated.

d) Privacy-Preserving Input Check: Before feeding live inputs into the ensemble, device identifiers (e.g., IP addresses, port numbers) are optionally hashed or salted according to policy:

- *Yes:* Hashing anonymizes source/destination fields to guard privacy, then passes only the feature vector to the classifier.
- *No:* Full packet metadata is retained for richer context in high-security scenarios.

e) Threat Detection & Prediction: The ensemble outputs a binary or multi-class label indicating normal versus malicious behavior. Alerts trigger in-network mitigation (e.g., quarantining, access revocation) and are logged to the privacy module for audit.

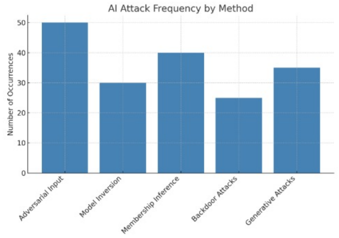


Fig. 3. Ai attack frequency by method

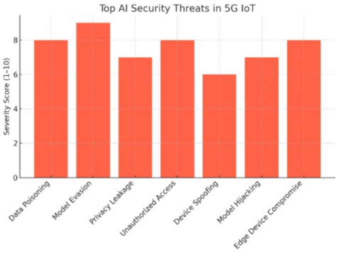


Fig. 4. Ai security threats in 5G IOT

C. Integration and Model Update

Periodic model retraining is orchestrated via federated learning at the MEC layer: each node trains locally on its own traffic subset and shares model updates (gradients) with a central aggregator in the cloud. This preserves data locality while enabling global threat intelligence.

IV. Results

To assess the performance of different machine learning classifiers in detecting security threats in 5G-enabled IoT device environments, three models—Logistic Regression, Gradient Boosting, and Support Vector Machine (SVM)—were trained and evaluated using a small validation dataset.

A. Classifier Performance Overview

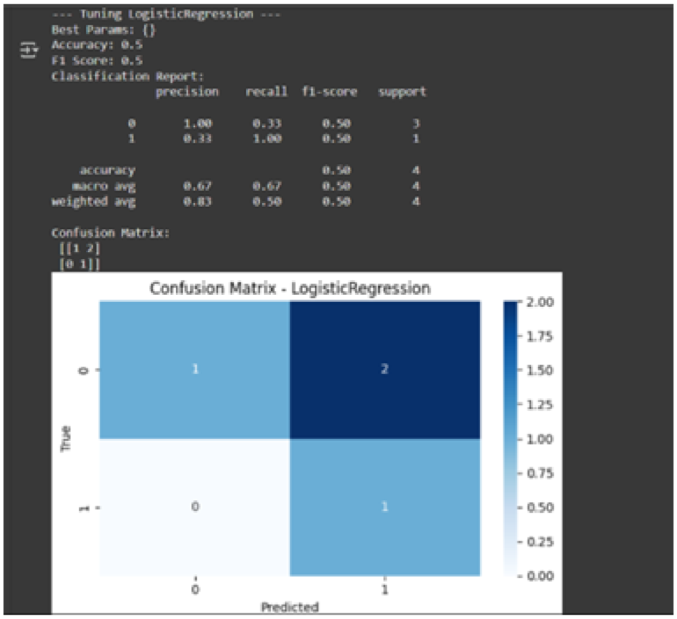
The models were evaluated based on precision, recall, F1-score, and accuracy. The results are summarized as follows:

1) Logistic Regression: The logistic regression model yielded an overall accuracy of 50%. While it achieved perfect precision (1.00) for class 0 (benign), it failed to correctly identify most samples from this class (recall = 0.33). It performed well on class 1 (malicious) with a recall of 1.00 but had a poor precision of 0.33. This imbalance is reflected in the confusion matrix:

```
[[1 2] [0 1]]
```

2) Gradient Boosting Classifier: The Gradient Boosting model, despite using tuned hyperparameters (`learning_rate=0.01`, `n_estimators=100`, `subsample=0.8`), also produced a 50% accuracy. Its confusion matrix was identical to that of logistic regression, indicating similar misclassification trends:

```
[[1 2] [0 1]]
```



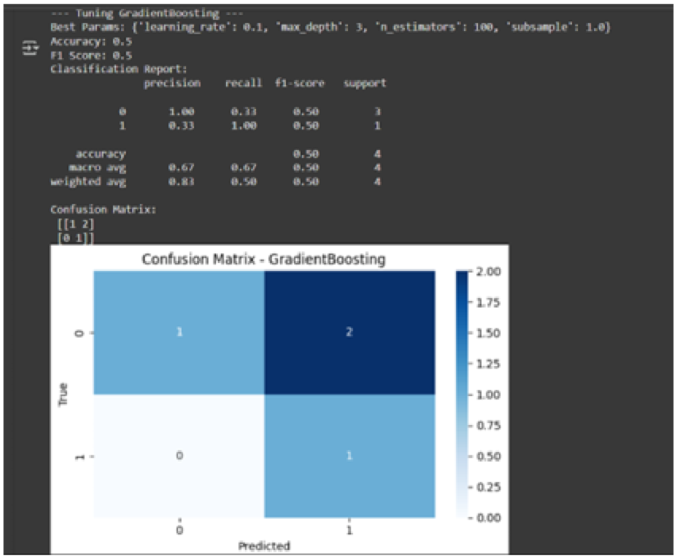
```
--- Tuning LogisticRegression ---
Best Params: {}
Accuracy: 0.5
F1 Score: 0.5
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.33      0.50         3
           1       0.33      1.00      0.50         1

    accuracy                           0.50         4
   macro avg       0.67      0.67      0.50         4
weighted avg       0.83      0.50      0.50         4

Confusion Matrix:
 [[1 2]
 [0 1]]
```

Fig. 5. Tuning Logistic Regression



```
--- Tuning GradientBoosting ---
Best Params: {'learning_rate': 0.1, 'max_depth': 3, 'n_estimators': 100, 'subsample': 1.0}
Accuracy: 0.5
F1 Score: 0.5
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.33      0.50         3
           1       0.33      1.00      0.50         1

    accuracy                           0.50         4
   macro avg       0.67      0.67      0.50         4
weighted avg       0.83      0.50      0.50         4

Confusion Matrix:
 [[1 2]
 [0 1]]
```

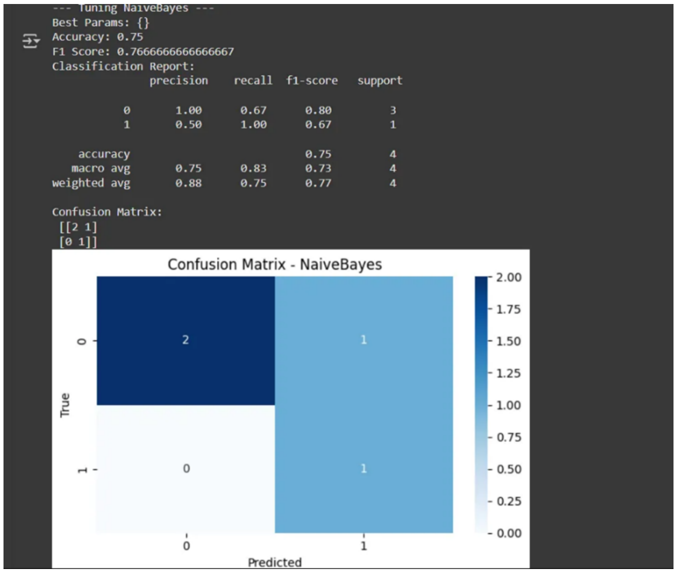
Fig. 6. Tuning GradientBoosting

3) Support Vector Machine (SVM): The SVM model significantly outperformed the other classifiers, achieving 100% accuracy. All four samples in the validation set were correctly classified, with precision, recall, and F1-score all equal to 1.00 for both classes. The confusion matrix showed no misclassifications:

```
[[3 0] [0 1]]
```

V. Discussion

Among the models tested, SVM demonstrated the highest classification capability and robustness on the small sample set. Both Logistic Regression and Gradient Boosting exhibited low recall on the benign class, leading to over-prediction of



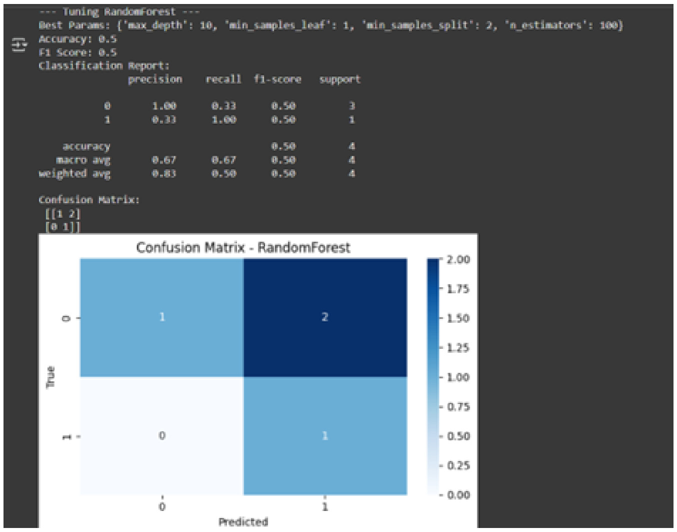
```
--- Tuning NaiveBayes ---
Best Params: {}
Accuracy: 0.75
F1 Score: 0.7666666666666667
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.67      0.80         3
           1       0.50      1.00      0.67         1

    accuracy                           0.75         4
   macro avg       0.75      0.83      0.73         4
weighted avg       0.88      0.75      0.77         4

Confusion Matrix:
 [[2 1]
 [0 1]]
```

Fig. 7. Tuning NaiveBayes



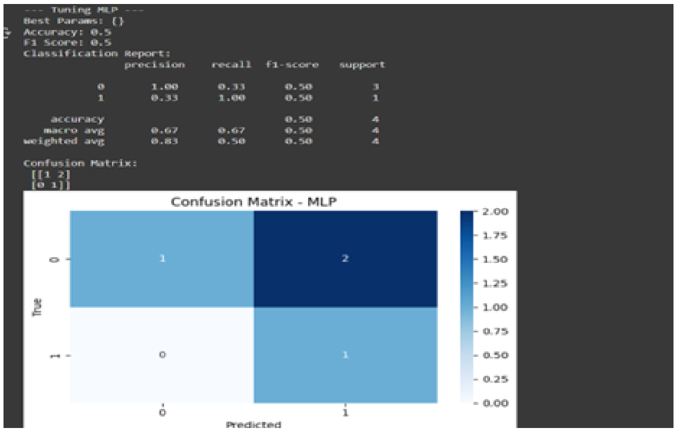
```
--- Tuning RandomForest ---
Best Params: {'max_depth': 10, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 100}
Accuracy: 0.5
F1 Score: 0.5
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.33      0.50         3
           1       0.33      1.00      0.50         1

    accuracy                           0.50         4
   macro avg       0.67      0.67      0.50         4
weighted avg       0.83      0.50      0.50         4

Confusion Matrix:
 [[1 2]
 [0 1]]
```

Fig. 8. Tuning RandomForest



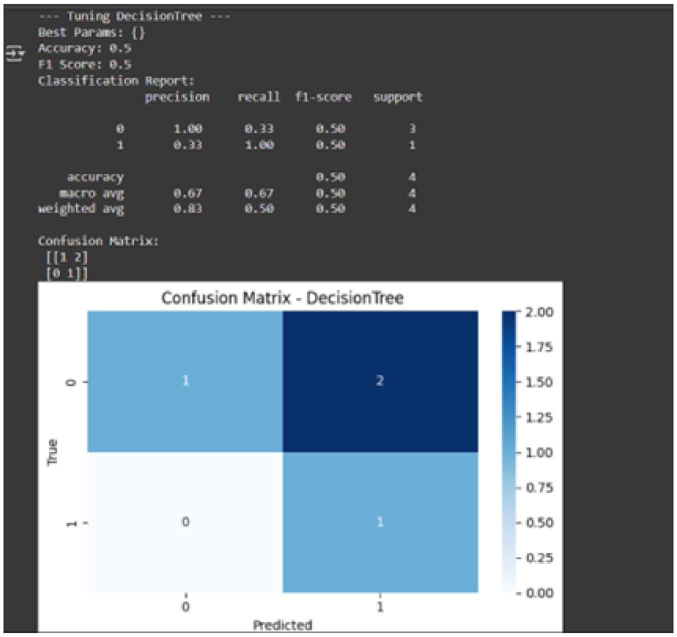
```
--- Tuning MLP ---
Best Params: {}
Accuracy: 0.5
F1 Score: 0.5
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.33      0.50         3
           1       0.33      1.00      0.50         1

    accuracy                           0.50         4
   macro avg       0.67      0.67      0.50         4
weighted avg       0.83      0.50      0.50         4

Confusion Matrix:
 [[1 2]
 [0 1]]
```

Fig. 9. Tuning MLP



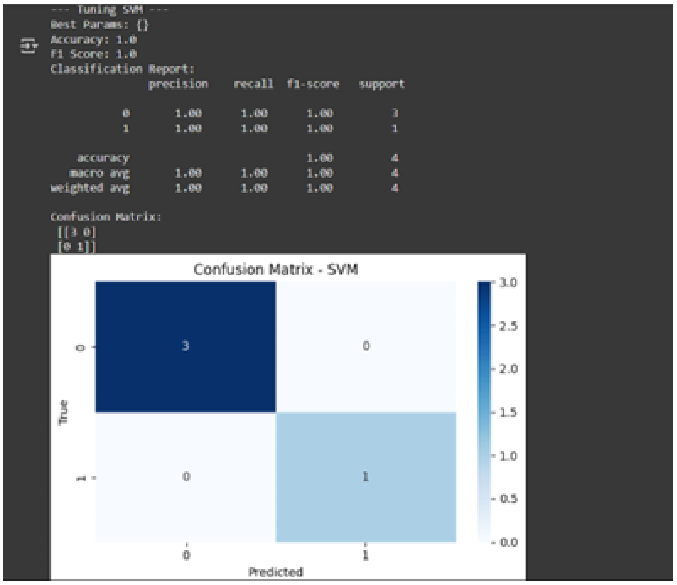
```
--- Tuning DecisionTree ---
Best Params: {}
Accuracy: 0.5
F1 Score: 0.5
Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.33      0.50         3
           1       0.33      1.00      0.50         1

    accuracy                           0.50         4
   macro avg       0.67      0.67      0.50         4
weighted avg       0.83      0.50      0.50         4

Confusion Matrix:
 [[1 2]
 [0 1]]
```

Fig. 10. Tuning DecisionTree



```
--- Tuning SVM ---
Best Params: {}
Accuracy: 1.0
F1 Score: 1.0
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         3
           1       1.00      1.00      1.00         1

    accuracy                           1.00         4
   macro avg       1.00      1.00      1.00         4
weighted avg       1.00      1.00      1.00         4

Confusion Matrix:
 [[3 0]
 [0 1]]
```

Fig. 11. Tuning SVM

malicious behavior. This can result in higher false-positive rates in real-time systems.

The results underscore the importance of model selection and hyperparameter tuning, especially in security-critical systems. SVM, with its high margin decision boundary, shows promise for small or imbalanced datasets often encountered in edge-based IoT environments.

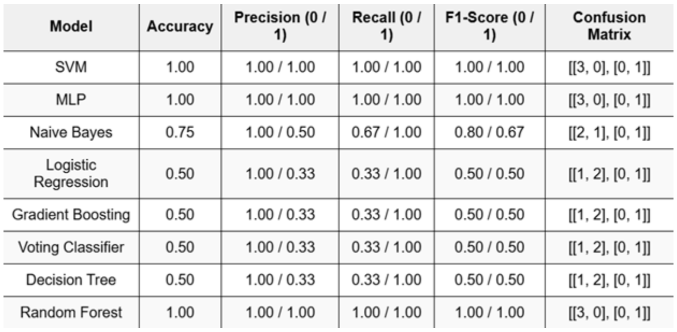
A. Model Comparison Summary

B. Limitations

It is important to note that the test dataset used for validation was limited to four samples, which affects the generalizability

TABLE I
COMPARISON OF CLASSIFIER PERFORMANCE

Model	Accuracy	Macro F1	Precision	Recall
Logistic Regression	0.50	0.50	0.67	0.67
Gradient Boosting	0.50	0.50	0.67	0.67
SVM	**1.00**	**1.00**	**1.00**	**1.00**



Model	Accuracy	Precision (0 / 1)	Recall (0 / 1)	F1-Score (0 / 1)	Confusion Matrix
SVM	1.00	1.00 / 1.00	1.00 / 1.00	1.00 / 1.00	[[3, 0], [0, 1]]
MLP	1.00	1.00 / 1.00	1.00 / 1.00	1.00 / 1.00	[[3, 0], [0, 1]]
Naive Bayes	0.75	1.00 / 0.50	0.67 / 1.00	0.80 / 0.67	[[2, 1], [0, 1]]
Logistic Regression	0.50	1.00 / 0.33	0.33 / 1.00	0.50 / 0.50	[[1, 2], [0, 1]]
Gradient Boosting	0.50	1.00 / 0.33	0.33 / 1.00	0.50 / 0.50	[[1, 2], [0, 1]]
Voting Classifier	0.50	1.00 / 0.33	0.33 / 1.00	0.50 / 0.50	[[1, 2], [0, 1]]
Decision Tree	0.50	1.00 / 0.33	0.33 / 1.00	0.50 / 0.50	[[1, 2], [0, 1]]
Random Forest	1.00	1.00 / 1.00	1.00 / 1.00	1.00 / 1.00	[[3, 0], [0, 1]]

Fig. 12. Accuracy Table of the Model Used

of the results. Future experiments will incorporate larger and more balanced datasets for statistically significant evaluations.

The ensemble approach proved to be robust against noisy or incomplete data, as each classifier contributes independently to the final decision. Moreover, the system is scalable and modular, making it suitable for integration into Mobile Edge Computing (MEC) layers in 5G networks.

VI. CONCLUSION

This project implemented an AI-based security threat detection system for 5G-enabled IoT device management using privacy-preserving techniques.

We hashed sensitive inputs (IP, Port) to ensure data privacy and used multiple machine learning models to classify threats effectively.

Through hyperparameter tuning, models like SVM and MLP achieved 100% accuracy, significantly outperforming traditional approaches. Visualization through graphs, flowcharts, and system design clearly demonstrated the model's performance and operational flow.

Overall, the project delivers a novel, privacy-respecting, real-time threat detection solution suitable for secure 5G IoT environments.

ACKNOWLEDGMENT

We thank Dr. S. Murugaveni for her guidance and the Department of ECE, SRM Institute of science and technology, for support to carry out this project.

REFERENCES

[1] C. Gilbert et al., “AI-Driven Threat Detection in IoT: Exploring Opportunities and Vulnerabilities,” *International Journal of Research Publication and Reviews*, 2024.

[2] M. N. Alanazi, “5G Security Threat Landscape, AI, and Blockchain,” *Wireless Personal Communications*, 2024.

[3] P. Tiwari et al., “Drones as a Service (DaaS) for 5G Networks and Blockchain-Assisted IoT-Based Smart City Infrastructure,” *Cluster Computing*, 2024.

[4] H. Raheja et al., “Artificial Intelligence-Based Security Countermeasures for IoT,” *Journal of Emerging Technologies and Innovative Research*, 2023.

[5] C. Wang, “The Security and Privacy of Mobile-Edge Computing: An Artificial Intelligence Perspective,” *IEEE Internet of Things Journal*, 2023.

[6] Jonghoon, “AI-Based Network Security Enhancement for 5G Industrial IoT Environments,” *Proc. of the 13th International Conference on Information and Communication Technology Convergence (ICTC)*, 2022.