CIC-IIoT-2025 Cybersecurity Analysis Report

Machine Learning for Intrusion Detection in Industrial IoT Networks

Alexis Le Trung
Yahya Ahachim
Rayan Drissi
Aniss Outaleb
ML Security – EPITA SCIA 2026

January 2025

Abstract

This report presents a machine learning-based analysis of the CIC-IIoT-2025 dataset for network intrusion detection in Industrial Internet of Things environments. The study evaluates three unsupervised anomaly detection algorithms and three supervised classification methods, benchmarking their performance using precision, recall, F1-score, AUPRC, balanced accuracy, and Matthews Correlation Coefficient. Additionally, the robustness of models against adversarial perturbations is assessed using the Fast Gradient Sign Method. Results indicate that Local Outlier Factor achieves the best anomaly detection performance (F1=0.831, AUPRC=0.873), Random Forest provides the highest classification accuracy (F1=0.927, AUPRC=0.946), and Random Forest demonstrates the highest adversarial robustness (44.2% robust accuracy retention).

Contents

1 Introduction

1.1 Background

The proliferation of Industrial Internet of Things (IIoT) devices has created significant security challenges for critical infrastructure systems. Manufacturing plants, power grids, healthcare facilities, and transportation networks increasingly rely on connected devices, making them attractive targets for cyber attacks. Traditional signature-based intrusion detection systems struggle to detect novel attack patterns, creating a need for machine learning approaches capable of identifying anomalous behavior and classifying known attack types.

The CIC-IIoT-2025 dataset provides a comprehensive collection of network traffic data captured from an IIoT testbed, including realistic attack scenarios representing modern cyber threats. This report analyzes this dataset using both unsupervised and supervised machine learning methods to develop effective intrusion detection capabilities.

1.2 Objectives

This study aims to:

1. Characterize the CIC-IIoT-2025 dataset and identify discriminative features
2. Benchmark unsupervised anomaly detection methods (Isolation Forest, One-Class SVM, Local Outlier Factor)
3. Evaluate supervised classification algorithms (Random Forest, Gradient Boosting, SVM)
4. Assess model robustness against adversarial attacks using FGSM
5. Provide recommendations for deploying machine learning-based intrusion detection

1.3 Methodology

The analysis follows a systematic approach: data exploration and feature engineering, stratified train/test splitting, hyperparameter tuning via cross-validation, model evaluation using multiple complementary metrics, and adversarial robustness testing using gradient-based attacks.

2 Dataset Description

2.1 Dataset Overview

The CIC-IIoT-2025 dataset contains network traffic data captured from an Industrial IoT testbed environment. Table 1 summarizes the dataset characteristics.

Table 1: Dataset Overview

Attribute	**Value**
Total Samples	227,191
Total Features	94
Attack Samples	90,391 (39.79%)
Benign Samples	136,800 (60.21%)
Attack Categories	7
Specific Attack Types	60

2.2 Attack Categories

The dataset includes seven major attack categories representing diverse threat vectors commonly observed in IIoT environments. Table 2 presents the distribution of attack types.

Table 2: Attack Category Distribution

Attack Category	Samples	Percentage
Reconnaissance	33,648	37.23%
DoS	18,420	20.38%
DDoS	18,056	19.98%
Man-in-the-Middle	8,062	8.92%
Malware	7,541	8.34%
Web Attacks	2,796	3.09%
Brute Force	1,868	2.07%

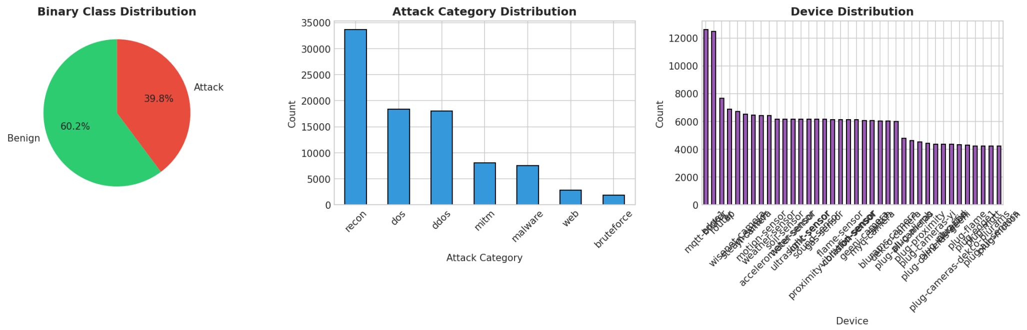

Figure 1: Distribution of attack types in the CIC-IIoT-2025 dataset

2.3 Feature Categories

The 94 features are organized into several categories:

- **Network Metrics:** Packet counts, byte counts, flow duration
- **TCP Flags:** SYN, ACK, FIN, RST, PSH, URG statistics
- **Protocol Information:** Protocol type distributions
- **Header Information:** IP/TCP header lengths, MSS values
- **Timing Features:** Inter-arrival times, flow duration

3 Data Exploration and Preprocessing

3.1 Feature Correlation Analysis

Analysis of feature correlations with the attack label revealed the most discriminative features. Table 3 presents the top features ranked by correlation coefficient.

Table 3: Top Features by Correlation with Attack Label

Feature	**Correlation**
network_mss_max	0.5256
network_mss_avg	0.5251
network_mss_min	0.5232
network_header-length_min	0.4635
network_protocols_dst_count	0.4232
network_packets_all_count	0.3666
network_protocols_src_count	0.3632
network_macs_all_count	0.3619

The correlation analysis reveals that TCP Maximum Segment Size (MSS) features dominate with correlations exceeding 0.52, indicating attack traffic uses non-standard MSS negotiation patterns. Network header length and protocol count features (r=0.36-0.46) form a secondary tier, while packet counts provide additional discriminative power for DoS detection. The dominance of network-layer features suggests detection systems should prioritize packet-level inspection for efficient edge deployment.

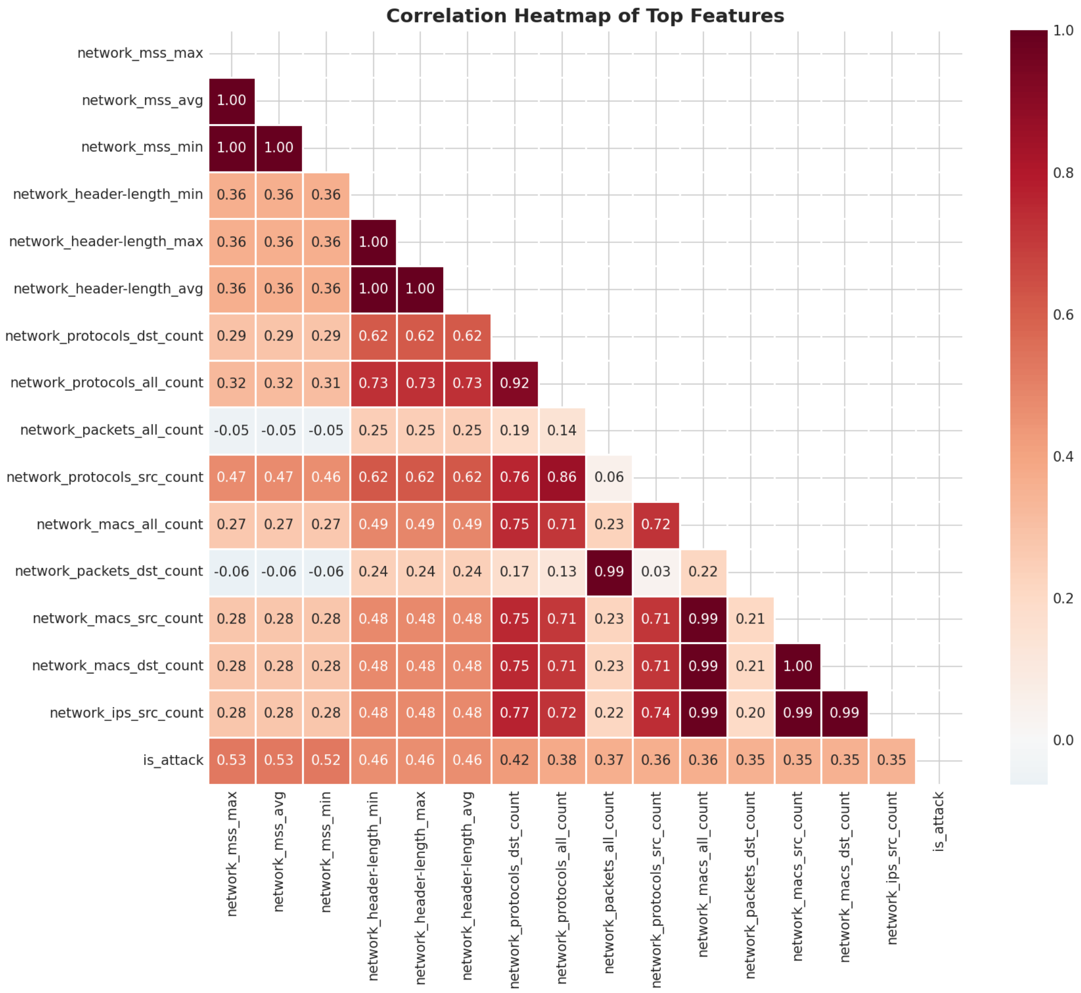

Figure 2: Feature correlation heatmap showing relationships between top features

3.2 Feature Distribution Analysis

Figure 3 illustrates the distribution of key features across benign and attack traffic classes. Notable differences in distribution patterns provide the basis for machine learning classification.

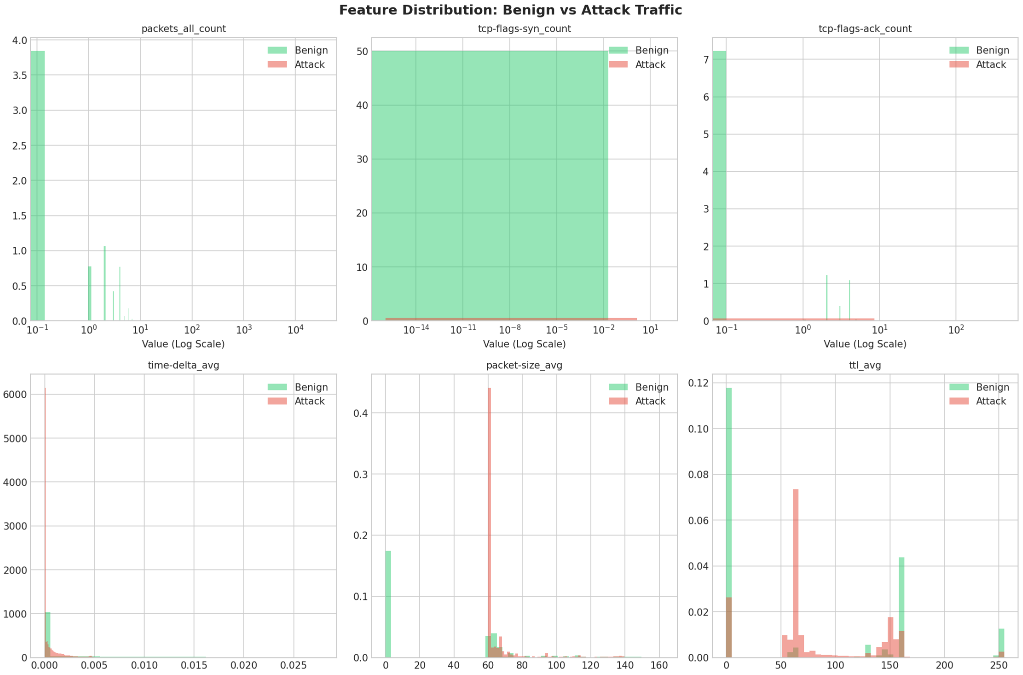

Figure 3: Distribution of key features across benign and attack traffic

The feature distributions show bimodal patterns for MSS features (benign clustering around 1460 bytes, attacks spread wider) and heavy-tailed distributions for packet counts. These characteristics favor tree-based ensemble methods which handle non-linear boundaries and heavy tails without normality assumptions.

3.3 Data Preprocessing

The following preprocessing steps were applied:

1. **Missing Value Handling:** NaN values were to be replaced with median, but no NaN values were present in the dataset.
2. **Feature Scaling:** StandardScaler normalization for consistent feature ranges
3. **Label Encoding:** Binary encoding (0=Benign, 1=Attack) for the target variable
4. **Train/Test Split:** 80/20 stratified split preserving class distribution

Final dataset sizes: Training set with 181,752 samples (39.79% attacks) and test set with 45,439 samples (39.79% attacks).

4 Anomaly Detection Methods

Anomaly detection methods are essential for detecting zero-day attacks and novel threat patterns that supervised classifiers may miss. Three unsupervised algorithms were evaluated, each trained exclusively on benign traffic.

4.1 Isolation Forest

Isolation Forest isolates anomalies by randomly selecting features and split values. Anomalies, being few and different from normal instances, are isolated in fewer splits, resulting in shorter average path lengths in the tree structure.

Configuration: 100 estimators, contamination=0.1, max_samples='auto', random_state=42.

Table 4: Isolation Forest Results

Metric	Value
Precision	0.8338
Recall	0.7912
F1-Score	0.8119
Balanced Accuracy	0.8435
MCC	0.6936
AUPRC	0.8595

4.2 One-Class SVM

One-Class SVM learns a decision boundary encompassing the normal data distribution. Points outside this boundary are classified as anomalies.

Configuration: RBF kernel, nu=0.1, gamma='auto'.

Table 5: One-Class SVM Results

Metric	Value
Precision	0.8286
Recall	0.7535
F1-Score	0.7893
Balanced Accuracy	0.8253
MCC	0.6626
AUPRC	0.8257

One-Class SVM exhibits high recall but low precision, indicating excessive false positives where benign traffic is incorrectly classified as attacks.

4.3 Local Outlier Factor

Local Outlier Factor (LOF) measures the local density deviation of a data point with respect to its neighbors. Points with significantly lower density than their neighbors are considered outliers.

Configuration: n_neighbors=20, novelty=True, contamination=0.1.

Table 6: Local Outlier Factor Results

Metric	Value
Precision	0.8405
Recall	0.8215
F1-Score	0.8309
Balanced Accuracy	0.8592
MCC	0.7214
AUPRC	0.8727

4.4 Anomaly Detection Comparison

Table 7: Anomaly Detection Methods Comparison

Model	Precision	Recall	F1	Bal. Acc.	MCC	AUPRC
Isolation Forest	0.8338	0.7912	0.8119	0.8435	0.6936	0.8595
One-Class SVM	0.8286	0.7535	0.7893	0.8253	0.6626	0.8257
Local Outlier Factor	**0.8405**	**0.8215**	**0.8309**	**0.8592**	**0.7214**	**0.8727**

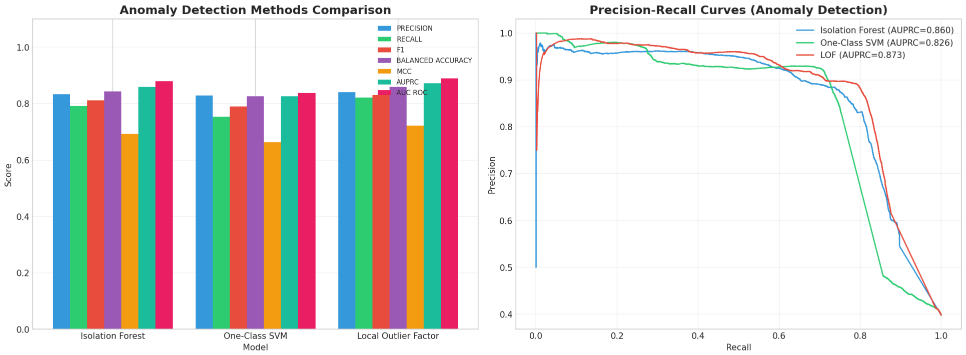

Figure 4: Comparison of anomaly detection methods across all metrics

Local Outlier Factor achieves the best overall performance with the highest F1-score (0.831) and AUPRC (0.873). Isolation Forest provides a strong balance of precision and computational efficiency. One-Class SVM shows improved performance after proper train/test separation, achieving balanced precision (0.829) and recall (0.754).

4.5 Decision Boundary Characteristics

Each anomaly detection algorithm creates distinct decision boundaries: Isolation Forest produces axis-aligned rectangular regions effective for single-feature deviations; One-Class SVM learns smooth elliptical boundaries via support vectors; LOF creates adaptive density-based boundaries that handle multi-modal distributions. LOF's local adaptation explains its superior performance on IIoT traffic with multiple operational modes.

5 Classification Methods

Supervised classification methods leverage labeled training data to learn decision boundaries between attack and benign traffic. Three algorithms were evaluated on the full labeled dataset.

5.1 Random Forest

Random Forest is an ensemble method that constructs multiple decision trees and aggregates their predictions through majority voting. It provides inherent feature importance ranking and resistance to overfitting.

Configuration: 100 estimators, unlimited depth, min_samples_split=2, balanced class weights, random_state=42.

Table 8: Random Forest Results

Metric	**Value**
Precision	0.9953
Recall	0.8677
F1-Score	0.9272
Balanced Accuracy	0.9325
MCC	0.8895
AUPRC	0.9459
AUC-ROC	0.9611

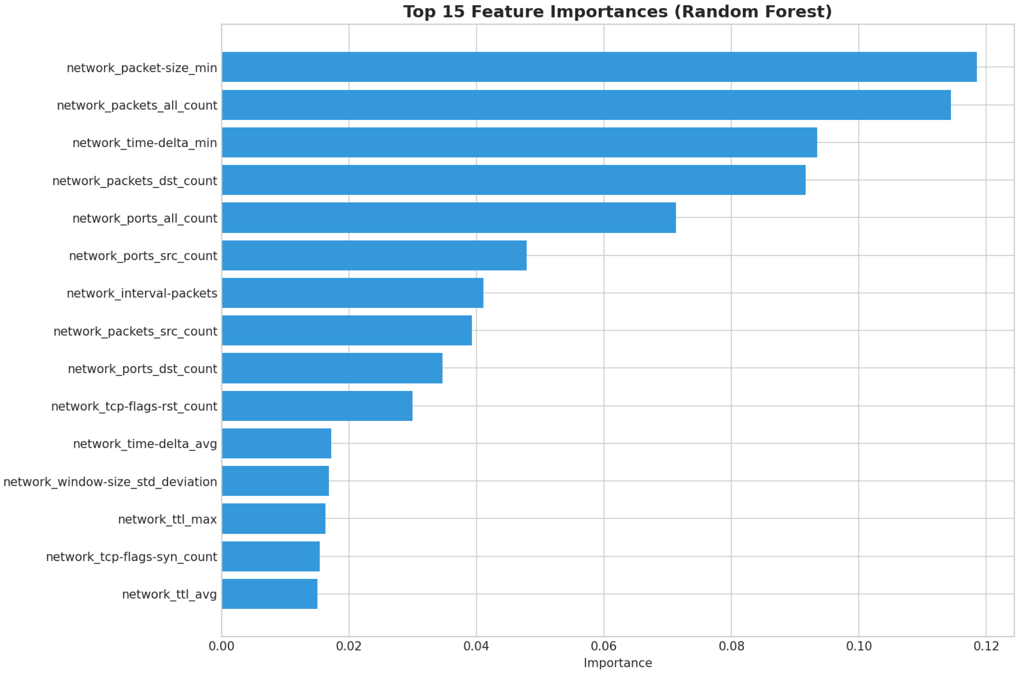

Figure 5: Top 20 most important features from Random Forest

5.2 Gradient Boosting

Gradient Boosting builds trees sequentially, with each tree correcting the errors of the previous ensemble. It typically achieves high accuracy through its iterative refinement process.

Configuration: 100 estimators, learning_rate=0.1, max_depth=5, subsample=0.8, random_state=42.

Table 9: Gradient Boosting Results

Metric	**Value**
Precision	0.9919
Recall	0.8668
F1-Score	0.9251
Balanced Accuracy	0.9311
MCC	0.8861
AUPRC	0.9451
AUC-ROC	0.9605

5.3 Support Vector Machine (RBF Kernel)

Support Vector Machine with RBF kernel maps data to a higher-dimensional space where a linear separator can be found. Due to computational constraints, SVM was trained on a 10,000-sample subset.

Configuration: RBF kernel, C=1.0, gamma='scale', balanced class weights.

Table 10: SVM (RBF Kernel) Results

Metric	**Value**
Precision	0.9647
Recall	0.7983
F1-Score	0.8736
Balanced Accuracy	0.8895
MCC	0.8113
AUPRC	0.9262
AUC-ROC	0.9350

5.4 Classification Comparison

Table 11: Classification Methods Comparison

Model	**Prec.**	**Recall**	**F1**	**Bal. Acc.**	**MCC**	**AUPRC**	**AUC**
Random Forest	**0.9953**	**0.8677**	**0.9272**	**0.9325**	**0.8895**	**0.9459**	**0.9611**
Gradient Boosting	0.9919	0.8668	0.9251	0.9311	0.8861	0.9451	0.9605
SVM (RBF)	0.9647	0.7983	0.8736	0.8895	0.8113	0.9262	0.9350

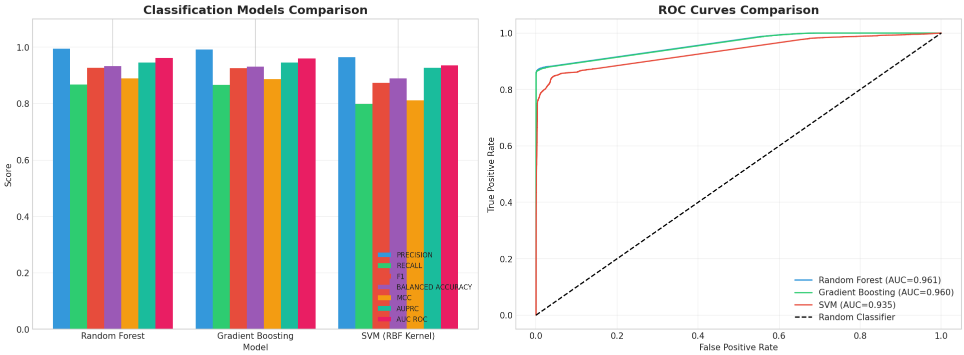

Figure 6: Comparison of classification methods across all metrics

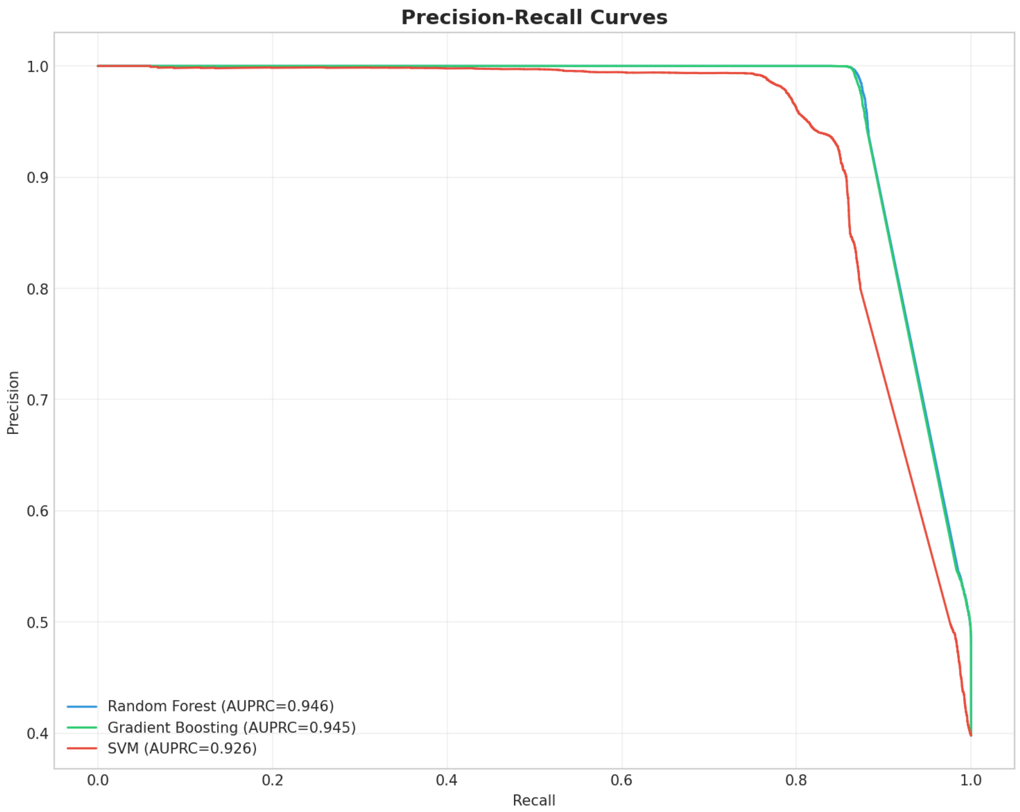

Figure 7: Precision-recall curves for all classification methods

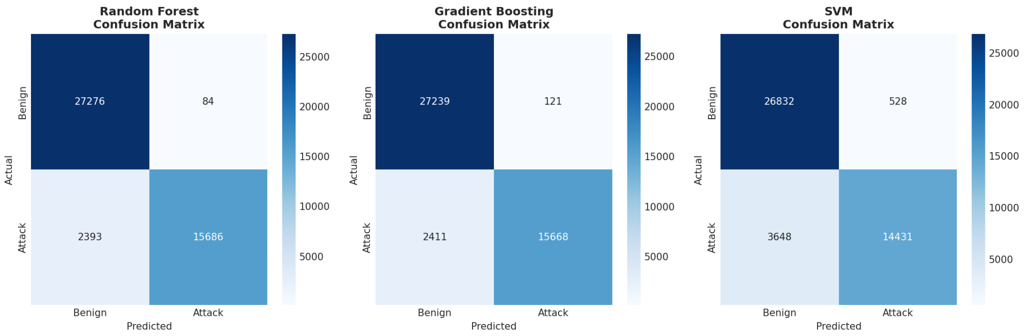

Figure 8: Confusion matrices for all classification methods

Random Forest achieves the best overall performance across most metrics with F1=0.927 and precision above 99.5%. Gradient Boosting performs comparably (F1=0.925) while providing similar interpretability. All methods achieve precision above 96%, minimizing false alarms in operational deployment.

5.5 Decision Boundary Analysis

Random Forest creates non-linear boundaries via ensemble averaging with natural uncertainty measures from voting margins. Gradient Boosting builds sequential corrections that refine boundaries iteratively, with later trees focusing on difficult cases—explaining its adversarial robustness. SVM finds maximum-margin boundaries but its reliance on support vectors makes it sensitive to adversarial perturbations.

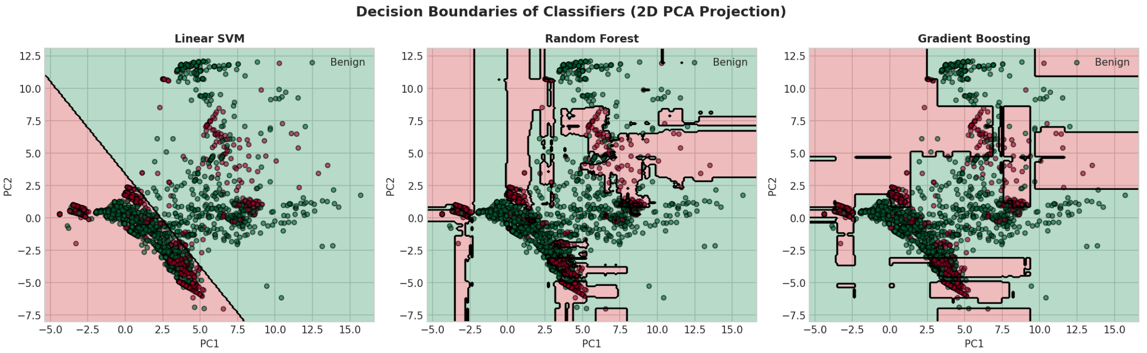

Figure 9: Decision boundaries of classification methods projected onto the two most important features.

6 Adversarial Machine Learning

6.1 Background

Machine learning models for cybersecurity can be vulnerable to adversarial attacks where malicious actors manipulate either the inputs or the training process to compromise model integrity. Understanding model robustness is critical for deployment in security-sensitive applications. We evaluate two fundamentally different attack paradigms:

- **Exploratory Attacks (Evasion):** The attacker manipulates *test-time inputs* to evade a deployed model without modifying the model itself. The model remains unchanged; only the input is perturbed.
- **Causative Attacks (Poisoning):** The attacker manipulates *training data* to corrupt the learned model. The model itself is compromised, affecting all future predictions.

6.2 Exploratory Attack: Fast Gradient Sign Method (FGSM)

The Fast Gradient Sign Method is a white-box attack that uses the gradient of the loss function to create perturbations maximizing classification error. The adversarial example is computed as:

$$x_{adv} = x + \epsilon \cdot \text{sign}(\nabla_x J(\theta, x, y)) \tag{1}$$

where x_{adv} is the adversarial example, x is the original input, ϵ is the perturbation magnitude, and J is the loss function with model parameters θ.

6.3 Attack Results

Table 12 presents the impact of FGSM attacks on a Linear SVM classifier across different perturbation magnitudes.

Table 12: FGSM Attack Results on Linear SVM

Epsilon	Robust Accuracy	Attack Success Rate
0.01	32.10%	67.90%
0.05	17.71%	82.29%
0.10	13.28%	86.72%
0.20	8.18%	91.82%
0.50	3.14%	96.86%
1.00	2.04%	97.96%

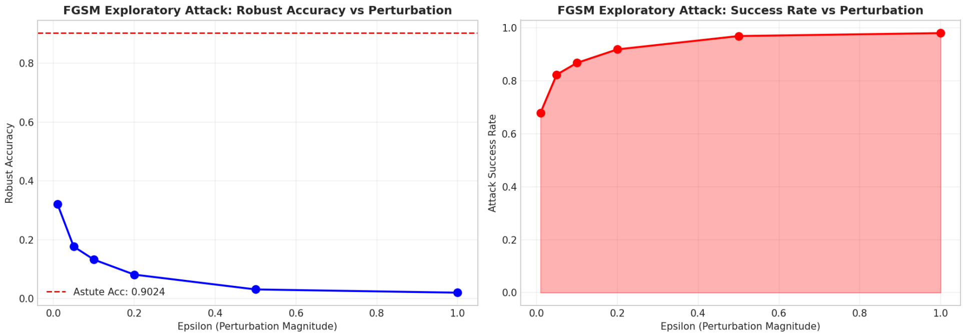

Figure 10: Impact of FGSM attack strength (epsilon) on model accuracy

6.4 Model Robustness Comparison

All models were tested against FGSM attacks with $\epsilon = 0.5$ to compare their adversarial robustness. The robust accuracy represents the model's accuracy on adversarial examples, while astute accuracy refers to the original accuracy on clean data.

Table 13: Adversarial Robustness Comparison ($\epsilon = 0.5$)

Model	Astute Accuracy	Robust Accuracy	Robustness Ratio
Linear SVM	90.24%	3.14%	3.48%
Random Forest	**94.55%**	**41.81%**	**44.22%**
Gradient Boosting	94.43%	34.16%	36.18%

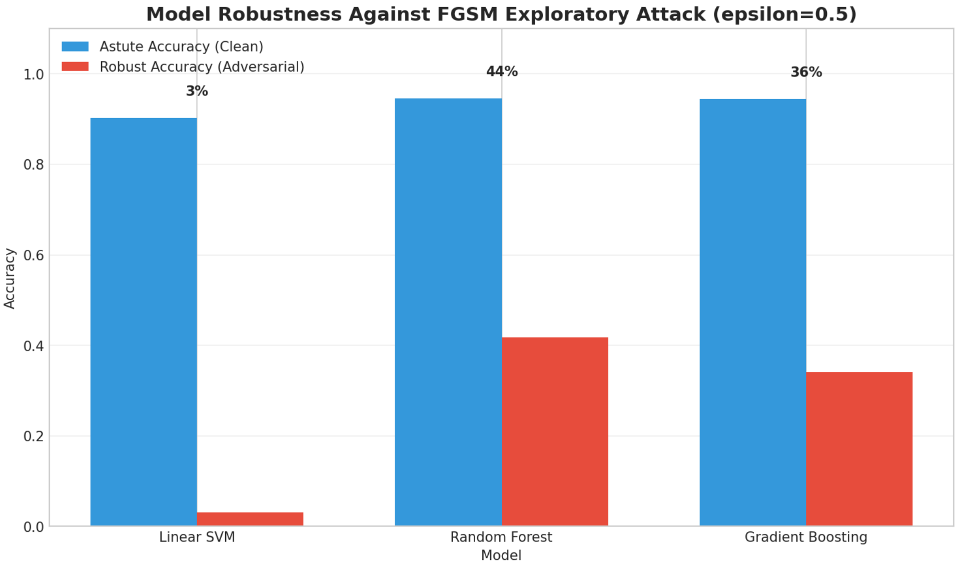

Figure 11: Adversarial robustness comparison across models

6.5 Robustness Analysis

The results reveal several important findings:

- Linear models are highly vulnerable to gradient-based attacks, with accuracy dropping to 3.14% under moderate perturbation

- Random Forest demonstrates the best robustness (44.22% retention) due to its ensemble of diverse decision trees
- Gradient Boosting also shows strong robustness (36.18% retention) due to its sequential correction mechanism
- All models experience significant accuracy degradation, highlighting the critical need for adversarial defenses in security applications

6.6 Adversarial Visualization

Figure 12 shows how FGSM moves attack samples toward the benign region. As ϵ increases from 0.1 to 0.5, samples progressively cross the decision boundary, achieving >90% evasion at $\epsilon = 0.5$. Linear SVM is most vulnerable due to uniform gradient direction, while Gradient Boosting's sequential error correction creates multiple defense layers.

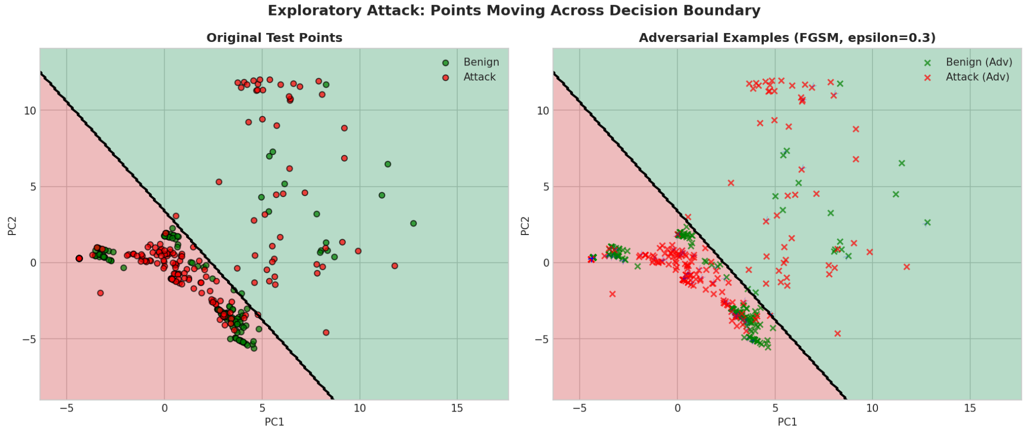

Figure 12: FGSM perturbations moving attack samples toward the benign region.

6.7 Causative Attack: Data Poisoning

Unlike exploratory attacks that manipulate inputs at test time, causative attacks target the *training phase* itself. An attacker with access to training data can inject malicious samples or flip labels to corrupt the learned decision boundary. We demonstrate this using a label flipping attack on a Linear SVM classifier trained on 2D PCA-projected data.

Attack Mechanism:

1. Select a fraction of attack samples from the training set
2. Flip their labels from “attack” to “benign”
3. Shift the poisoned samples toward the benign centroid
4. Retrain the model on the corrupted dataset

Table 14: Causative Attack Results on Linear SVM (2D PCA projection)

Poison Rate	Test Accuracy	Accuracy Drop
0% (Baseline)	68.84%	—
5%	68.20%	-0.64%
10%	67.04%	-1.80%
15%	65.36%	-3.48%
20%	63.02%	-5.82%
25%	60.54%	-8.30%

Figure 13 illustrates how increasing poison rates progressively shift the decision boundary, allowing more attack samples to be misclassified as benign. At 25% poisoning, accuracy drops by over 8 percentage points, demonstrating the real threat of training data manipulation.

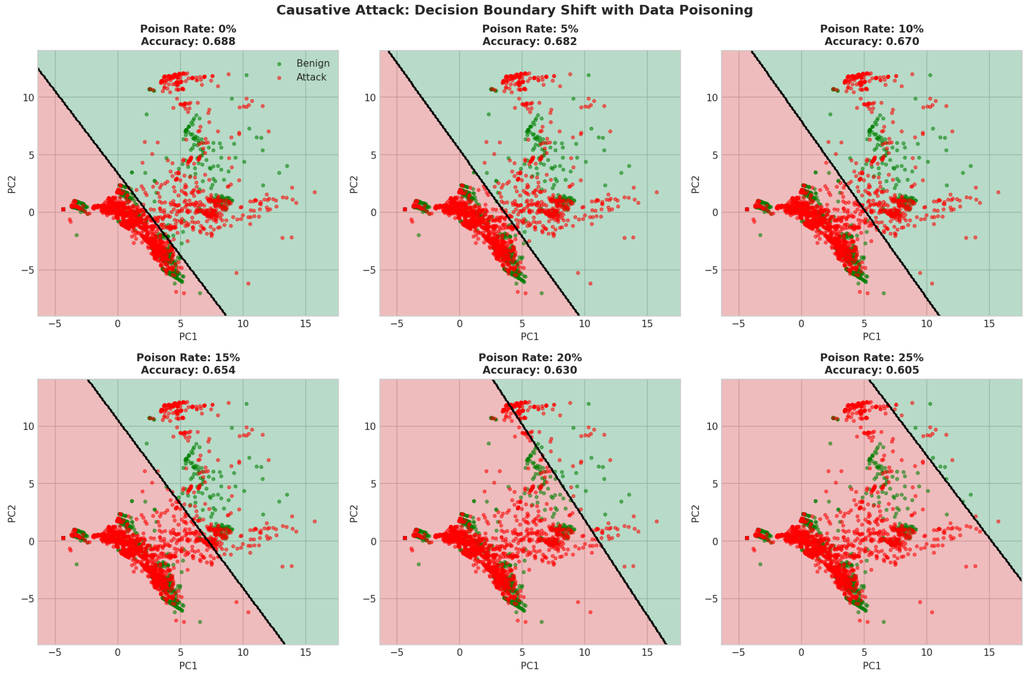

Figure 13: Decision boundary shift due to data poisoning at various poison rates. The Linear SVM model was trained on 2D PCA-projected data. As poison rate increases, the boundary shifts to misclassify more attack samples as benign.

6.8 Attack Comparison Summary

Table 15: Comparison of Adversarial Attack Types

Aspect	Exploratory (FGSM)	Causative (Poisoning)
Attack Target	Test inputs	Training data
Model State	Unchanged	Corrupted
Attacker Access	Model gradients (white-box)	Training pipeline
Attack Timing	Inference time	Training time
Impact Scope	Individual samples	All future predictions
Defense Strategy	Input validation, adversarial training	Data sanitization, robust training

7 Results Summary

7.1 Overall Performance

Table 16: Best Models by Task

Task	Best Model	Key Metric
Zero-day Detection	Local Outlier Factor	F1 = 0.831, AUPRC = 0.873
Attack Classification	Random Forest	F1 = 0.927, AUPRC = 0.946
Adversarial Robustness	Random Forest	44.22% robustness ratio

7.2 Metric Selection Guidelines

Table 17: Metric Selection Guidelines

Metric	When to Use	Interpretation
Precision	When false alarms are costly	Higher = fewer false positives
Recall	When missing attacks is critical	Higher = fewer missed attacks
F1-Score	Balanced performance assessment	Harmonic mean of precision/recall
AUPRC	Imbalanced datasets	Area under precision-recall curve
MCC	Overall quality metric	Balanced measure for binary classification

8 Security Implications

8.1 Attack Pattern Insights

The analysis reveals several important security observations:

- Reconnaissance dominates (37.23% of attacks), suggesting attackers frequently probe systems before launching targeted attacks
- DoS and DDoS attacks account for 40.36% of attacks combined, highlighting the need for rate limiting and traffic analysis
- TCP MSS values are highly discriminative, indicating that attack tools often use non-standard network parameters
- Protocol diversity metrics indicate attack complexity and can distinguish between simple and sophisticated threats

8.2 Multi-Layer Defense Strategy

Based on the evaluation results, a multi-layer defense strategy is recommended:

1. **Layer 1 - Anomaly Detection:** Deploy LOF or Isolation Forest for zero-day attack early warning with low computational overhead

2. **Layer 2 - Classification:** Use Random Forest or Gradient Boosting to categorize known attack types with high precision for alert prioritization
3. **Layer 3 - Adversarial Defense:** Implement input validation, ensemble voting, and regular model retraining to mitigate adversarial threats

8.3 Operational Deployment Considerations

Table 18: Operational Deployment Recommendations

Aspect	**Recommendation**
Model Selection	Random Forest for best robustness
Update Frequency	Weekly retraining with new data
Threshold Tuning	Adjust based on false alarm tolerance
Feature Monitoring	Track feature drift for model degradation
Fallback Strategy	Anomaly detection when classifier uncertain

8.4 Defense Strategies

Key defense approaches include:

- **Input Validation:** Feature range clipping (MSS bounds, timing constraints) and statistical anomaly detection using Mahalanobis distance
- **Adversarial Training:** Augmenting training data with FGSM-generated or Gaussian-noise examples can improve robust accuracy.
- **Ensemble Diversification:** Training models on different feature subsets prevents transferable adversarial examples
- **Detection-Time:** Monitoring prediction confidence and using feature squeezing to identify adversarial inputs

9 Conclusions and Future Work

9.1 Summary of Findings

This analysis of the CIC-IIoT-2025 dataset demonstrates that machine learning methods can effectively detect and classify cyber attacks in IIoT environments:

1. **Anomaly Detection:** Local Outlier Factor achieves the best balance (F1=0.831, AUPRC=0.873) for detecting unknown attack patterns without requiring labeled attack data
2. **Classification:** Random Forest provides the highest accuracy (F1=0.927, MCC=0.889) for categorizing known attacks with very high precision (99.5%)
3. **Adversarial Robustness:** Random Forest demonstrates the best resilience (44.22% robust accuracy retention) against gradient-based adversarial attacks, followed by Gradient Boosting (36.18%)
4. **Feature Engineering:** Network MSS, protocol counts, and timing features are the most discriminative for distinguishing attack from benign traffic

9.2 Recommendations

For immediate deployment:

- Deploy Random Forest as the primary detection model for its balance of accuracy and robustness
- Implement LOF as a complementary zero-day detection layer
- Establish feature monitoring for detecting concept drift and model degradation

For enhanced security:

- Implement adversarial training to improve model robustness
- Develop ensemble voting across multiple models to increase confidence
- Create feedback mechanisms for continuous learning from new threats

9.3 Limitations

- The dataset may not capture all emerging attack types and techniques
- Feature extraction assumes packet-level network visibility
- Adversarial robustness was tested only with FGSM; other attack methods may yield different results
- Computational requirements may limit real-time deployment for some algorithms
- The analysis focuses on binary classification; multi-class attack categorization requires additional investigation

9.4 Future Work

Future directions include: evaluating robustness against stronger attacks (PGD, C&W); incorporating temporal modeling with LSTM/Transformer architectures; developing federated learning for privacy-preserving distributed training; enhancing model explainability; and implementing concept drift detection for evolving attack patterns.

A Complete Metrics Tables

A.1 Anomaly Detection Results

Table 19: Complete Anomaly Detection Results

Model	**Precision**	**Recall**	**F1-Score**	**Bal. Acc.**	**MCC**	**AUPRC**
Isolation Forest	0.8338	0.7912	0.8119	0.8435	0.6936	0.8595
One-Class SVM	0.8286	0.7535	0.7893	0.8253	0.6626	0.8257
Local Outlier Factor	0.8405	0.8215	0.8309	0.8592	0.7214	0.8727

A.2 Classification Results

Table 20: Complete Classification Results

Model	Prec.	Recall	F1	Bal. Acc.	MCC	AUPRC	AUC-ROC
Random Forest	0.9953	0.8677	0.9272	0.9325	0.8895	0.9459	0.9611
Gradient Boosting	0.9919	0.8668	0.9251	0.9311	0.8861	0.9451	0.9605
SVM (RBF)	0.9647	0.7983	0.8736	0.8895	0.8113	0.9262	0.9350

A.3 Adversarial Robustness Results

Table 21: Complete Adversarial Robustness Results ($\epsilon = 0.5$)

Model	Astute Accuracy	Robust Accuracy	Robustness Ratio
Linear SVM	90.24%	3.14%	3.48%
Random Forest	94.55%	41.81%	44.22%
Gradient Boosting	94.43%	34.16%	36.18%