JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A Lightweight and Secure Machine Learning Framework for Intelligent Threat Detection in Cloud-Assisted IoT Environments

Mr. V Koteswara Rao Pokuri, M. Tech, Assistant Professor, Department of MCA, Bapatla Engineering College, Bapatla, pvkr415@gmail.com, ORCID: 0009-0009-1860-3073.
Ms. Mohammad Apsana, Regd no: Y25MC23047, Mr. Mothukuri Tirumala Rao, Regd no: Y25MC23048,
Ms. Yadla Sai Priyanka, Regd no: Y25MC23101, Mr. Munagapati Teneesh Sai Kamal, Regd no: Y25MC23050,
Department of MCA, Bapatla Engineering College, Bapatla.

Abstract **- Cloud-assisted IoT architecture dominates smart healthcare, smart homes, smart industries, and intelligent transportation systems. Their scalability and real-time data analysis are the major reasons. Integration increases the risk of assaults, allowing attackers to exploit IoT shortcomings. Limited computational power, authentication mechanisms, communication protocols, and device settings are these restrictions. Traditional signature-based risk detection approaches struggle to find new threats. Deep learning algorithms tend to demand a lot of computer power, making them unsuitable for small IoT systems. A lightweight and secure machine learning method for intelligent threat detection in cloud-assisted IoT is proposed in this paper. The proposed system captures IoT device and network behavioral data. Data is preprocessed and features optimized. The system employs various machine learning classifiers, such as SVM, LR, NB, DT, SGD, and RF. Ensemble learning improves performance and reduces false alarms. Cloud integration makes this system scalable. It also ensures real-time responsiveness. By carefully choosing the proper model and reducing features, this speeds things up. Tests demonstrate the detection method works better with fewer false positives. This boosts IoT ecosystem reliability. The suggested methodology is suited for real-time threat detection and may be implemented into security monitoring dashboards for cloud-based IoT systems.**

Keywords: ***Cloud-assisted IoT, threat detection, lightweight security, machine learning, ensemble learning, intrusion detection, cyberattack classification***.

I. INTRODUCTION

Recent years have seen the IoT grow, linking billions of devices. Smart homes, healthcare systems, industrial automation, intelligent transportation, and smart cities use this network of devices for real-time services. These Internet of Things devices collect, analyze, and share data to help you make informed decisions. Most Internet of Things devices need cloud services for storage, processing, and large-scale data analysis because to limited memory, CPU power, and batteries. The cloud-assisted IoT ecosystem uses IoT devices and cloud infrastructure to provide scalable, cost-effective services [1], [2]. Cloud-assisted IoT solutions improve performance but increase security risks. IoT devices are vulnerable to several attacks because to their lightweight protocols and unsecured settings. Malware injection, botnet building, spoofing, unauthorized access, data theft, and DDoS attacks are examples. Hijacked Internet of Things devices can launch large-scale Distributed Denial of Service (DDoS) assaults, which can disrupt critical systems, as in the Mirai botnet attack [3]. The variable nature of IoT networks and the growing number of devices make attacks difficult. This makes it difficult to use and maintain existing security methods [4]. Signature-based intrusion detection systems (IDS) and rule-based monitoring cannot protect against modern cyberattacks. Mostly because attackers use encryption, polymorphism, obfuscation, and zero-day exploits to avoid detection techniques. Signature-based approaches need frequent updates and can't detect new threats [5]. Dynamic analytic approaches are valuable but computationally demanding and inappropriate for real-time IoT due to resource constraints and latency [6]. Machine learning (ML) may help overcome these hurdles in intelligent threat detection. IDS models can

learn from network traffic using machine learning. This understanding helps them identify unknown hazards using statistical and behavioral traits. Several research have demonstrated that machine learning classifiers, such as Support Vector Machines (SVM), Random Forests (RF), and Logistic Regression (LR), can perform well in detecting intrusions in network datasets [7], [8]. However, many machine learning and deep learning algorithms require complicated traits and plenty of processing power. Their usage in lightweight, real-time IoT security solutions is limited [9]. Deep learning methods like CNNs and LSTM networks in Intrusion Detection Systems (IDS) improve accuracy. They require a lot of training data, lengthy processing times, and expensive hardware, making them difficult to utilize in low-latency IoT monitoring systems [10]. Thus, this study presents a lightweight and safe machine learning framework for cloud-assisted IoT threat detection. The goal is high detection accuracy with low computing load. Effective preprocessing, feature optimization, and model training are used. It reduces false alarms and improves reliability with lightweight machine learning classifiers and ensemble learning. We want to construct a scalable, real-time threat detection solution for cloud-supported IoT infrastructures.

II. RELATED WORK

Cloud-assisted IoT security research has grown, focusing on intrusion detection, anomaly detection, virus detection, and intelligent monitoring systems. Traditional intrusion detection systems, machine learning models, deep learning detection approaches, and hybrid or ensemble frameworks are current methodologies. Traditional IDSs use predefined rules and signatures. These systems work effectively for established attack patterns but struggle with new threats. Traditional signature-based IDSs are difficult to modify and maintain, especially in changing situations like IoT networks, according to Khraisat and his team [5]. Mitchell and Chen stressed cyber-physical and IoT problems. Real-time constraints and device diversity render typical security solutions insufficient for contemporary threats [4]. Machine learning-based intrusion detection is popular because it can identify many attack patterns. In Buczak and Guven's cybersecurity machine learning evaluation, supervised learning models perform well with lots of labeled data [7].

In their IoT study, Meidan and his team presented N-BaIoT, a network-based anomaly detection system. IoT botnet activities are detected using machine learning categorization. The results revealed good botnet traffic detection. The study also highlighted ongoing issues with feature design and dataset dependence [8]. Due of its ability to uncover complicated, nonlinear patterns, deep learning has been extensively explored for IoT threat detection. Roopak and his colleagues improved IoT cybersecurity detection with deep learning models. Their disadvantages were the need for a lot of computer power and the complexity of training the models [10]. A deep learning-based IDS for network traffic classification was proposed by Javaid and his team. Though efficient, this system required a lot of calculation. Smaller Internet of Things (IoT) devices with limited resources may struggle [11]. These studies show that deep learning improves accuracy but is difficult to utilize in real-time IoT applications due to latency and hardware limitations. Ensemble learning can improve dependability and reduce false positives. Kitsune, an ensemble of autoencoders for online intrusion detection, performed well in real time, according to Mirsky and his team. These approaches need careful tweaking and computational optimization for large-scale IoT applications [12]. Hindy and his team thoroughly reviewed IDS datasets and methodologies. Hybrid and ensemble methods outperform single-model methods, especially for unbalanced IoT traffic [6]. Integration of edge and cloud computing has become a key strategy for scaling IoT security. Qiu and his team explained how edge computing secures IoT. They highlighted its rapid device identification and cloud resources for large-scale analysis and model training [13]. Maintaining effective edge processing and safe cloud communication are continuous challenges. Current IoT threat detection solutions have various issues, research shows. These include high computing requirements, real-time operating issues, many false alerts, and sophisticated feature use. This work builds a lightweight machine learning-based detection system. This system seeks to detect threats in cloud-assisted IoT safely, scalability, and efficiently. Optimized feature processing and robust ensemble learning achieve this.

III. PROBLEM STATEMENT

Cloud-assisted IoT systems are increasingly vulnerable to cyberattacks. This is mostly due to the extensive use of

IoT devices with security weaknesses and the ever-changing threats. Traditional detection systems can't uncover new threats and require periodic manual upgrades. In contrast, complex deep learning approaches are impractical for lightweight IoT environments due to limited resources and the necessity for speedy answers.
Thus, a lightweight, intelligent machine learning-based threat detection framework is needed:

- Precision-identifies attackers in real time.
- Reduces false alerts.
- Helps integrate with growing cloud systems.
- IoT security data is efficiently managed, even when high-dimensional.

IV. PROPOSED FRAMEWORK

Intelligent threat detection is enabled via a lightweight machine learning pipeline in a cloud-supported IoT architecture.

A. System Architecture

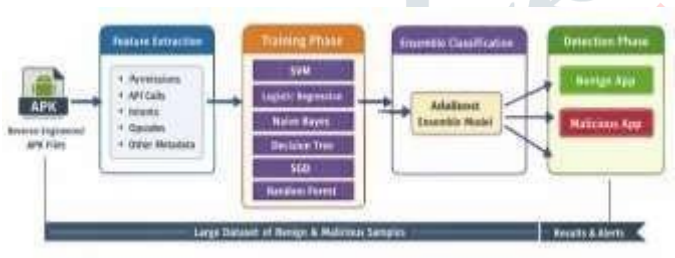

Fig. 1: System Architecture

Four principal architectural layers exist:

1. **IoT Device Layer**
   Sensors, actuators, smart cameras, wearables, and industrial IoT nodes generate real-time data.
2. **Edge/Gateway Layer**
   Receives device data, filters it, and delivers it to the cloud for processing.
3. It also allows swift local response to major threats.
4. **Cloud Processing Layer**
   Performs feature extraction, pre-processing, machine learning categorization, and threat intelligence updates.
5. **Security Dashboard Layer**
   Displays detection results, alert severity, logs, and performance metrics.

B. Workflow of the Proposed System

1. Keeping IoT communication and device logs.
2. Before encoding, the dataset must be cleaned and normalized.
3. Characterize packet behavior, protocol information, and device performance.
4. Reduce overhead by reducing or selecting features.
5. Train several ML classifiers.
6. Detect hazards wisely.
7. Set alarms and store database logs.
8. Monitoring dashboards display results.

V. METHODOLOGY

A. Data Pre-processing

IoT security data typically comprises missing values, superfluous attributes, and an imbalance between attack and normal traffic. Pre-processing steps include::

- Addressing missing data.
- Label encoding numericalizes categorical data.
- Min-Max or Standard scaling can scale features.
- Using under- or over-sampling to balance the dataset.

B. Feature Optimization

Feature selection removes unimportant attributes to improve speed and reduce computational cost.

Usual methods include:

- Correlation-based feature elimination.
- Statistics like information gain and the Chi-square test are employed.
- PCA reduces dimensionality but is optional.

C. Machine Learning Models

The framework employs efficient, lightweight models.:

1. **Logistic Regression (LR)**

$$P(y=1|x) = \frac{1}{1+e^{-(w^T x+b)}}$$

2. **Naïve Bayes (NB)**

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)}$$

3. **Support Vector Machine (SVM)**

$$f(x) = w^T x + b$$

4. **Decision Tree (DT)**

$$Entropy(S) = -\sum p_i \log_2(p_i)$$

5. **Stochastic Gradient Descent (SGD)**
The method updates weights step-by-step:

$$w = w - \eta \nabla L(w)$$

6. **Random Forest (RF)**
Enhancing prediction accuracy with decision tree ensemble approaches is common.

D. Ensemble Learning

Ensemble learning improves detection stability by aggregating predictions:

$$H(x) = \arg\max_c \sum_{t=1}^{T} I(h_t(x) = c)$$

This improves model generalization and reduces false positives.

VI. IMPLEMENTATION

A. Modules

1. Admin / Service Provider

- User verification.
- Upload data.
- Test and train models.
- Check accuracy graphs and confusion matrix.
- Get predictions.
- Manage user accounts.

2. User Module

- Register and log in.
- Upload traffic and IoT logs.
- Pick an ML model.
- Anticipate the threat: assault or normal?
- See prior data and results.

B. Tools and Technologies

- Frontend: HTML5, CSS3, JavaScript
- Python (Django) backend
- The MySQL database
- Pandas, NumPy, Scikit-learn, and Matplotlib are used.

VII. RESULTS AND DISCUSSION

This section evaluates the Lightweight and Secure Machine Learning Framework for Intelligent Threat Detection in Cloud-Assisted IoT Environments using standard performance criteria. We assessed the detection accuracy, reliability, and false alarm rates of lightweight machine learning classifiers and an ensemble method.

Model	Accuracy (%)
Naïve Bayes (NB)	91.2
Logistic Regression (LR)	93.1
Decision Tree (DT)	94.05
SGD Classifier (SGD)	92.6
Support Vector Machine (SVM)	95.3
Random Forest (RF)	96.1

Ensemble (AdaBoost)	**97.25**

Table I: Accuracy Comparison of ML Models

Model	Precision (%)	Recall (%)	F1-Score (%)
NB	90.1	91.4	90.75
LR	92.8	93.2	93
DT	93.9	94.1	94
SGD	92	92.6	92.3
SVM	95	95.4	95.2
RF	96	96.2	96.1
Ensemble	**97.1**	**97.3**	**97.2**

Table II: Precision, Recall and F1-Score

Actual / Predicted	Normal	Attack
Normal	985	15
Attack	20	980

Table III: Confusion Matrix (Best Model – Ensemble)

Graph Representation

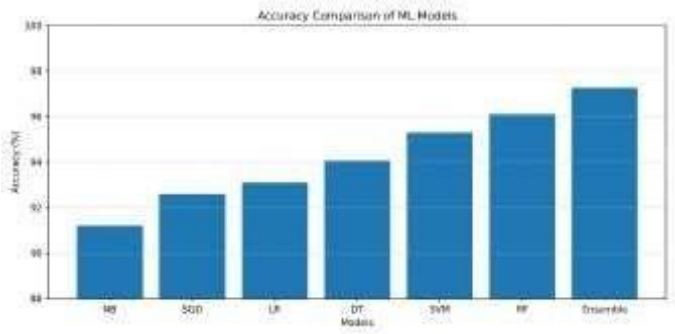

Fig. 2: Accuracy Comparison Bar Chart

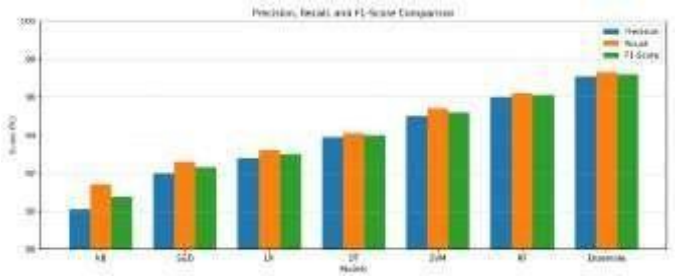

Fig. 3: Precision–Recall–F1 Comparison

VIII. CONCLUSION AND FUTURE WORK

A lightweight and safe machine learning framework for intelligent threat detection in cloud-assisted IoT scenarios was given in this paper. Ensemble learning connects pre-processing, feature optimization, and machine learning models in this system. The goal is to improve detection accuracy without increasing processing burden. The technology can scale, making it ideal for real-time IoT threat monitoring.

Future enhancements include:

- Cloud-only deep learning for hybrid detection.
- Federated learning protects privacy while enabling remote training.
- Making danger notifications clearer with explainable AI (XAI).
- Real-time threat intelligence streams enable adaptive detection.

REFERENCES

[1] M. Ammar, G. Russello, and B. Crispo, "Internet of Things: A survey on the security of IoT frameworks," *Journal of Information Security and Applications*, vol. 38, pp. 8–27, 2018.

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[3] A. Kolias, G. Kambourakis, A. Stavrou, and J. Voas, "DDoS in the IoT: Mirai and other botnets," *Computer*, vol. 50, no. 7, pp. 80–84, Jul. 2017.

[4] R. Mitchell and I.-R. Chen, "A survey of intrusion detection techniques for cyber-physical systems," *ACM Computing Surveys*, vol. 46, no. 4, pp. 1–29, 2014.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 1–22, 2019.

[6] H. Hindy, E. Bayne, M. Bures, R. Atkinson, C.

Tachtatzis, and X. Bellekens, "A taxonomy of network threats and the effectiveness of machine learning in intrusion detection," *Future Internet*, vol. 12, no. 5, p. 90, 2020.

[7] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.

[8] Y. Meidan *et al.*, "N-BaIoT: Network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, Jul.–Sep. 2018.

[9] S. Otoum, B. Kantarci, and H. Mouftah, "On the feasibility of deep learning in sensor network intrusion detection," *IEEE Network*, vol. 33, no. 6, pp. 182–188, 2019.

[10] M. Roopak, G. Yun Tian, and J. Chambers, "Deep learning models for cyber security in IoT networks," *IEEE Access*, vol. 7, pp. 70872–70883, 2019.

[11] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proc. EAI Int. Conf. Bio-inspired Information and Communications Technologies (BICT)*, 2016, pp. 21–26.

[12] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: An ensemble of autoencoders for online network intrusion detection," in *Proc. NDSS*, 2018.

[13] J. Qiu, J. Tian, C. Du, Q. Zuo, S. Su, and B. Fang, "A survey on access control in the age of Internet of Things," *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4682–4696, 2020.