Adaptive Cyberattack Detection in IoT-Edge-Cloud Environments Using Decision Tree Regressor

G. C. Shwethashree
Department of Computer Science and Engineering, Sri Jayachamarajendra College of Engineering, JSS Science and Technology University, Mysore, Karnataka, India
shwethashreegcresearch@gmail.com (corresponding author)

S. Manjula
Department of Computer Science and Engineering, Sri Jayachamarajendra College of Engineering, JSS Science and Technology University, Mysore, Karnataka, India
thejasyashas@sjce.ac.in

Received: 28 March 2025 | Revised: 27 May 2025 | Accepted: 1 June 2025

 DOI: https://doi.org/10.48084/etasr.11184

ABSTRACT

By facilitating smooth communication among smart devices, edge computing, and cloud environments, the Internet of Things (IoT) has reshaped several sectors. However, IoT networks are highly vulnerable to cyberattacks, particularly link attacks, which compromise security. In the last ten years, various existing Machine Learning (ML) and Deep Learning (DL) approaches have been presented for attack detection, but they often fail to maintain high accuracy when attack patterns evolve. This study proposes a Decision Tree Regressor (DTR) model for attack prediction and adaptation in an IoT Edge-Cloud environment. The model is implemented using SENSORIA for IoT data collection and CloudSim for edge-cloud simulation to efficiently detect attacks. The DTR model dynamically adapts to changes in attack behavior through statistical monitoring. The model was evaluated on the ToN-IoT and UNSW-NB15 datasets, achieving 99.92% and 99.96% accuracy, respectively, significantly outperforming existing approaches. The results demonstrate that DTR improves the accuracy of attack detection while adapting to evolving attack patterns, ensuring robust IoT security.

Keywords-IoT security; edge-cloud computing; cyberattack detection; link attacks; machine learning; decision tree regressor; attack adaptation

I. INTRODUCTION

The Internet of Things (IoT) has transformed current technology by facilitating smooth connections among intelligent devices, sensors, and computing infrastructures. IoT devices, such as smartwatches, industrial sensors, smart homes, and automobiles, are continuously collecting enormous volumes of data from their environment [1]. These devices generate real-time data that is transmitted for further analysis and processing. Edge computing manages the collected data effectively by analyzing it near the source, minimizing latency, and improving decision-making in real time [2]. However, when the edge nodes become overloaded or require additional computational resources, the data is transferred to cloud servers for more extensive processing and storage [3]. Dynamic interaction between IoT devices, edge computing, and cloud infrastructure ensures smooth operation across various domains, such as industrial automation, smart cities, and healthcare. Despite its advantages, the IoT ecosystem faces significant security challenges, particularly in the communication layer, i.e., the IoT-Edge and Edge-Cloud networks [4]. Most IoT applications, sensors, and devices operate with minimal memory and power, making them vulnerable to cyberattacks. Attackers target these IoT applications/sensors/devices using different methods, such as Domain Name Service (DNS), Man-in-the-Middle (MiTM), and Distributed Denial of Service (DDoS) attacks [5]. Among these, link attacks pose a severe threat by overwhelming IoT devices and edge servers that have large amounts of malicious traffic, leading to service disruptions and data breaches [6]. Given the critical role of IoT in various applications, detecting and mitigating such attacks is of utmost importance to ensure data integrity, system reliability, and user privacy.

Over the years, researchers have proposed several attack detection techniques to enhance IoT security. Many of these approaches leverage Machine Learning (ML) and Deep Learning (DL) models to detect attacks in IoT-Edge and Edge-Cloud networks. In [7], a lightweight approach employed Genetic Algorithm (GA) and the t-test for DDoS Attack

Detection (GADAD), to detect low- and high-volume DDoS attacks in IoT environments. GADAD employed an edge-based environment and was divided into three stages. The first included creating and preprocessing low- and high-volume DDoS attacks in an IoT environment, the second included the utilization of GA with statistical variables (GAStats) to select the best features for attack detection, and the third used different ML approaches, i.e., K-Nearest-Neighbor (KNN), Logistic-Regression (LR), Adaptive-Boosting (AB), Extra-Trees (ET), and Random-Forest (RF), to detect attacks in the collected low- and high-volume data. The ToN-IoT dataset was used for a real-time attack detection scenario. The results showed that 96% accuracy was achieved with GADAD-RF and 95% was achieved with GADAD-ET.

In [8], a hybrid DL model combined a Convolutional Neural Network (CNN) and Long-Short-Term Memory (LSTM) to detect DDoS attacks. In addition, ten ML approaches were used for the same purpose. Evaluations were carried out on two datasets, namely ToN-IoT and CICIoT2023, and the findings showed that the CNN-LSTM model achieved 98.75% and 99.995% accuracy on ToN-IoT and CICIoT2023, respectively. In [9], an approach was presented to detect attacks on the IoT fog-cloud layer using Federated Learning (FL). Additionally, the Split Learning (SL) concept was used to efficiently learn attack patterns. Ten ML approaches, including Artificial Neural-Network (ANN), Support-Vector Machine (SVM), Extreme-Gradient-Boosting (XGB), LSTM, CNN, RF, DT, LR, and NB, along with FL and SL, were evaluated on the NSL-KDD and UNSWB-NB15 datasets. The results showed that SL achieved higher accuracy, namely 99.23% and 98.02% for the NSL-KDD and UNSWB-NB15 datasets, respectively.

In [10], a DL approach called Deep-Ensemble Learning using Pruning approach (DEEPShield) was proposed to detect low- and high-volume DDoS attacks. The ensemble model used CNN-LSTM to analyze traffic in IoT devices/nodes (DEEPShield-Ensemble). Pruning was applied to refine CNN-LSTM, which helped in the deployment of the model in the IoT-Edge layer (DEEPShield-Pruning-Ensemble). This study also collected low- and high-volume DDoS attack datasets for attack detection. Furthermore, the model was tested on other datasets, including ISCX-12, CICIDS-2017, and ToN-IoT. In [11], an LSTM Residual Network (LSTM-ResNet) was used for attack detection, built on the basis of a Temporal-Convolutional-Residual (TCR) module. In the TCR module, a modified attention model was presented for feature selection/extraction. Evaluations were carried out using two datasets, UNSW-NB15 and ToN-IoT, where the model achieved 89.23% and 99.24% accuracy, respectively.

In [12], Bidirectional Gated Recurrent Unit (Bi-GRU) and LSTM were combined with softmax for the IoT-Edge layer in a smart agriculture environment. For faster training, the Bi-GRU-LSTM model incorporated the Truncated-Back-Propagation Through-Time (TBPTT) method for training long sequences. The Bi-GRU-LSTM was tested in three datasets, Edge-IIoT, ToN-IoT, and CICIDS2018, achieving accuracies of 98.32%, 99.55% and 99.82%, respectively. In [13], a DL approach was based on Graph Learning, called Graph-Learning Data-Link Anomaly Detection (GLDAE), which considered communication network and link attacks. The GLDAE includes a graph-enhancing module, a link-feature auto-encoder to extract features, a structured auto-encoder, and a discriminator for learning edge-related features and creating a latent graph-based structure. To enhance generalization for detecting attacks in different scenarios, contrastive learning was employed on the original and modified graphs. A module integrated structural and feature embeddings for the decoder to achieve joint learning among graph and edge-selected features. Finally, MLP was used to predict and classify attacks. NetFlow datasets, i.e., NF-ToN-IoT-v2, NG-ToN-IoT, NF-UNSW-NB15-v2, and NF-UNSW-NB15, were used to evaluate the model, achieving 98.84%, 97.67%, 96.52%, and 99.09% accuracies, respectively.

Previous studies on attack detection employed ML and DL techniques to achieve high accuracy, and most of them focused on either the UNSWB-NB-15 or the ToN-IoT dataset. However, these approaches struggled to maintain accuracy as attack patterns evolve over time. Many approaches fail to adapt dynamically, leading to a decrease in attack detection performance and increased FPs. To address these problems, this work presents an attack detection approach using a Decision Tree Regressor (DTR) model, designed to analyze the behavior of the IoT device and detect potential DDoS attacks with improved accuracy. By leveraging statistical monitoring and adaptive learning techniques, the proposed solution detects changes in attack patterns dynamically, ensuring the timely mitigation of security threats. The model continuously evaluates IoT data at the edge layer, and if significant attack behavior is detected, it detects the attack. Integration of DTR into the IoT Edge-Cloud framework enhances IoT security by providing a robust mechanism to efficiently detect and respond to link attacks. The contributions of this study are as follows:

- Proposes the DTR model for detecting link-based attacks in IoT Edge-Cloud environments.
- Implements a statistical monitoring technique to dynamically detect shifts in attack behavior and update the attack detection model accordingly.
- Enhances real-time attack detection by integrating the DTR model within an edge computing framework, reducing latency in threat detection.
- Improves accuracy and adaptability compared to existing ML- and DL-based approaches, ensuring better detection of evolving cyber threats.
- Provides a scalable security solution that balances computational efficiency while securing IoT communication between edge and cloud layers.

II. METHODOLOGY

To predict attacks using the DTR model, consider IoT collected data, denoted X, that changes its behavior with respect to time t:

$$X = \{(x_1, y_1), (x_2, y_2), \dots (x_t, y_t)\} \quad (1)$$

where x_t denotes a feature-vector for every t^{th} observation and $y_t \in \{0,1\}$, with $y_t = 1$ denoting attack and $y_t = 0$ denoting

normal behavior. The DTR model F maps input features X to get output y:

$$F = f(X) = \sum_{k=1}^{K} \theta_k \cdot h_k(X) \quad (2)$$

where K denotes the total number of DTR trees present, θ_k denotes weights assigned to k^{th} tree and $h_k(X)$ denotes the output of the k^{th} for input X. The main aim of DTR is to minimize log-loss while incorporating a regularization term for preventing overfitting. The loss function is given by:

$$L = \sum_{t=1}^{N} [y_t \log(p_t) + (1 - y_t) \log(1 - p_t)] + \sum_{k=1}^{K} \Omega(h_k) \quad (3)$$

where $p_t = \sigma\big(f(x_t)\big)$ is the predicted attack probability. The sigmoid-activation σ is given in (4) and $\Omega(h_k)$ is given in (5):

$$\sigma(x_t) = \frac{1}{1+e^{-x_t}} \quad (4)$$

$$\Omega(h_k) = \gamma N + \frac{1}{2}\lambda \big||w_k|\big|^2 \quad (5)$$

where e^{-x_t} denotes the exponential value of x_t, N denotes the number of leaves in each tree, w_k denotes the weight given to each leaf, and γ and λ are hyperparameters that control the tree complexity. The DTR model is trained on the IoT dataset by reducing L, optimizing the best tree structure using node splits, optimizing the best leaf weight w_k, and balancing accuracy and generalizing $\Omega(h_k)$. As attacks change patterns over time, this work uses statistical monitoring to detect shifts in attack behavior. Hence, at every time-step t, the statistical difference between new data x_t and initial data x_1 is measured using K-L divergence [14]:

$$X_{KL}(x_t || x_1) = \sum_{x \in X} x_t(x) \log \frac{x_t(x)}{x_1(x)} \quad (6)$$

where a high X_{KL} indicates significant changes in attack distribution. This approach updates the time window W of attack dynamically as $W = W \cup X_{KL}$. For updating W, the time window W is split into two segments, where the first consists of stable historical data S and a subwindow S_{sub} extracted from S, and the second consists of new testing data T, where $|T| = n$, such that $|S| \gg |T|$, where n is the cardinality of testing data T. In this approach, a t-test is used to check if the mean values of S and T are different, similar to the test presented in [7]. The t-test is evaluated using:

$$t - test = \frac{\bar{S} - \bar{T} - (\mu_1 - \mu_2 - \delta)}{\sigma_w \sqrt{(\frac{1}{n_1} + \frac{1}{n_2})}} \sim t(n_1 + n_2 - 2) \quad (7)$$

where $\bar{S}$ and $\bar{\bar{T}}$ are sample means of S and T, respectively, σ_w comprises of sample variance of S and T, i.e., σ_s and σ_T, and μ_1 and μ_2 are the hypotheses for the dynamic parameter δ. If a difference is detected using $t - test$, it means that attack behavior has changed. Hence, T is further divided into T_1 and T_2 test windows to analyze a specific time boundary t^* where the change (attack) has occurred. If change is detected, DTR is again retrained by resetting W as $W = \emptyset$ and $x_1 = x_t$ using the updated data. If no change is detected, DTR remains unchanged. Once trained, DTR predicts attacks using new IoT observations X_{new}:

$$F = f(X_{new}) = \sum_{k=1}^{K} \theta_k \cdot h_k(X_{new}) \quad (8)$$

The attack probability for a new observation is evaluated using:

$$p_{attack} = \sigma\big(f(X_{new})\big) \quad (9)$$

If $p_{attack} > threshold$, then $y = 1$ means that an attack was detected, else $y = 0$ denotes normal behavior. Figure 1 presents the complete process of the DTR model.

The novelty of this work is that it proposes a dynamic statistical monitoring system that identifies distributional changes in IoT data streams by continuously measuring K-L-divergence at each time step using (6). When significant change is detected using the hypothesis test (t-test) (7), this approach automatically isolates the segment where the shift occurred, records the boundary time t^*, and retrains the DTR model on the updated attack profile. This adaptive retraining mechanism ensures that the model evolves with changing attack behaviors, rather than remaining static or requiring manual intervention. Also, this approach introduces a rigorous data windowing strategy, splitting data into stable S and test T segments with dynamic resizing to retain historical consistency while being sensitive to recent shifts. These mechanisms are integrated into a complete decision loop, shown in Figure 1, enabling continuous online learning and high attack prediction accuracy in the face of adversarial dynamics.

III. RESULTS AND DISCUSSION

To validate the effectiveness of the DTR-based adaptive attack detection approach, a dynamic experimentation pipeline was implemented to capture attack behavior shifts on the UNSW-NB15 and ToN-IoT datasets. The key innovation lies in statistical monitoring and adaptive retraining strategy, which enables the DTR model to maintain high accuracy in dynamic and evolving IoT attack environments. Figure 1 shows the experimental flow of the architecture, where each incoming IoT observation from both datasets was subjected to KL-divergence-based drift detection (6). Once distributional drift was detected using the t-test (7), the observation window was segmented into historical S and T parts, and the boundary time t^* was recorded to identify the point of behavioral shift. The DTR model was then retrained using updated data starting from t^*, ensuring that the DTR model is adapted to the new attack patterns without manual intervention.

Moreover, the entire framework was deployed in a hybrid IoT-Edge-Cloud simulation environment. The SENSORIA simulator [15] was used to generate IoT-layer data streams using the UNSW-NB15 and ToN-IoT datasets, which were then processed at the edge using CloudSim-based infrastructure [16]. This setup mimics realistic multilayered IoT systems, where low-latency detection is crucial. For each dataset, the incoming data was processed in time steps to simulate a streaming environment. Unlike existing models that train once and assume static data distributions, this approach continuously monitors and adapts to evolving threats. The DTR model was dynamically re-trained based on statistically significant changes, making the proposed approach fundamentally different from traditional static or batch-trained models. The simulation was carried out on an Intel i7 processor and 16 GB RAM, ensuring efficient execution of the DTR model.

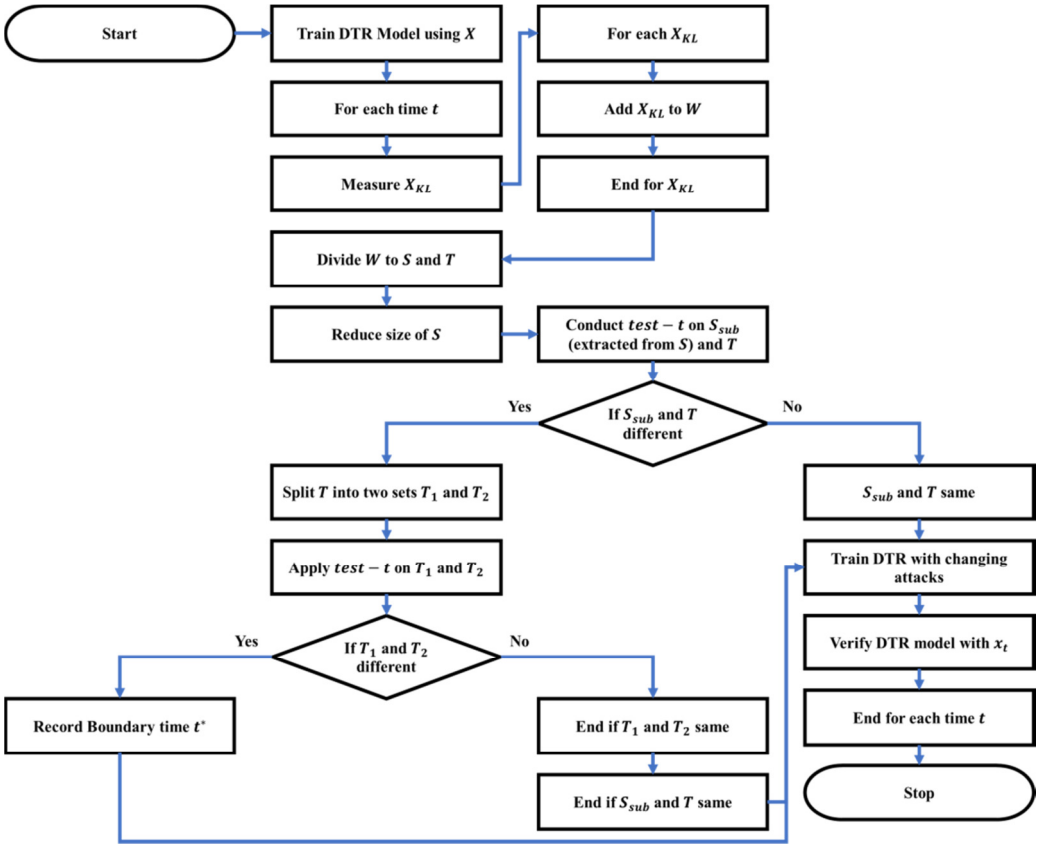

Fig. 1. Flow diagram of DTR.

A. Dataset

This study used the UNSW-NB15 [17-21] and ToN-IoT [22-29] datasets. The UNSW-NB15 dataset was created by the Australian Cyber Security Center (ACCS) and is an extensive database for attack detection. The dataset comprises a combination of standard and harmful network traffic, produced utilizing the IXIA PerfectStorm tool, and features nine distinct attack classes, including exploits, backdoors, worms, and DoS. The dataset features 49 attributes, providing diverse network behaviors for cybersecurity research. The ToN-IoT (Telemetry, Operational, and Network IoT) dataset is an extension of UNSW-NB15 that incorporates telemetry, network, and operating system data from IoT environments and includes real-world attack scenarios across multiple IoT services.

B. Performance Metrics

The following performance metrics were used to evaluate the DTR model.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

$$\text{Precision} = TP/(TP+FP) \quad (11)$$

$$\text{Recall} = TP/(TP+FN) \quad (12)$$

$$\text{F}-\text{score} = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (13)$$

C. Performance Evaluation

Figure 2 presents the performance of the DTR model on the UNSW-NB15 dataset, where it achieved 99.96% accuracy. In addition, the model achieved 99.97%, 99.96%, and 99.91% scores for precision, recall, and F-score. These results show that DTR was able to detect attacks having very few FPs and without missing any attacks. Moreover, the DTR shows reliability and robustness for attack detection because of the statistical test (7), which improves attack detection accuracy.

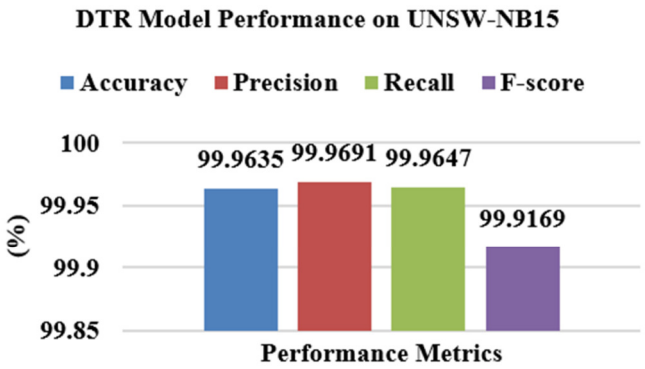

Fig. 2. DTR Performance Evaluation on the UNSW-NB15 dataset.

Figure 3 presents the performance of the DTR model on the ToN-IoT dataset, where it achieved 99.92% accuracy. In addition, the model achieved 99.88%, 99.87%, and 99.87% scores for precision, recall, and F-score. These results show that DTR was able to detect attacks having very few FPs and without missing any attacks. Furthermore, the DTR shows reliability and robustness for attack detection due to the statistical test (7), which improves the accuracy of attack detection. The results show that DTR was able to detect attacks, even when they evolved over time, providing a robust approach to secure IoT Edge-Cloud environments.

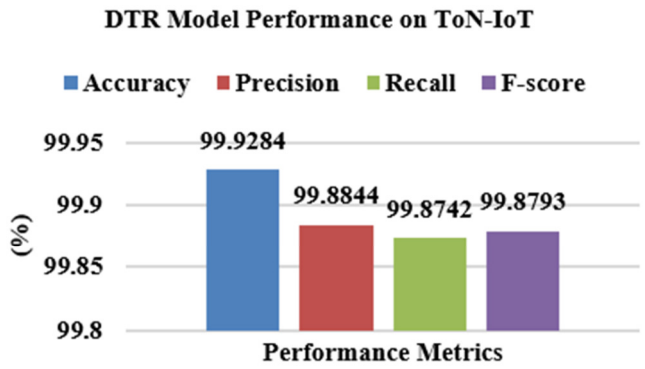

Fig. 3. DTR performance evaluation on the ToN-IoT dataset.

D. Comparative Study

The proposed DTR model was compared with existing approaches presented in the literature survey. Table I presents the comparative results of existing attack detection/detection approaches compared with the DTR model for the UNSW-NB15 dataset. Compared to the SL [9] and GLDAE [13] models, which achieved accuracies of 98.02% and 98.84%, respectively, the proposed DTR model achieved better accuracy at 99.96%, demonstrating a better ability to detect cyber threats. Additionally, the DTR achieved the highest precision (99.97%), recall (99.96%), and F-score (99.91%), outperforming LSTM-ResNet [11], which showed significantly lower accuracy (89.23%).

TABLE I. PERFORMANCE COMPARISON WITH EXISTING APPROACHES USING UNSW-NB15 DATASET

Ref.	Model	Accuracy	Precision	Recall	F-score
[9]	SL	98.02	98.12	98.02	98.11
[11]	LSTM-ResNet	89.23	883.83	87.77	88.25
[13]	GLDAE	98.84	98.8	98.84	98.78
Proposed	**DTR**	**99.9635**	**99.9691**	**99.9647**	**99.9169**

Table II compares the performance of the DTR model with existing methods presented in the literature survey for the ToN-IoT dataset. The findings show that the DTR model achieved 99.92% accuracy, outperforming GADAD-ET [7] (95%), CNN-LSTM [8] (98.75%), and GLDAE [13] (97.67%). Although DEEPShield-Ensemble [10] and DEEPShield-Pruning-Ensemble [10] achieved 99% accuracy, the DTR model surpassed them with improved precision (99.88%), recall (99.97%), and F-score (99.87%). Furthermore, compared to Bi-GRU-LSTM [12], which achieved 99.55% accuracy, DTR provided a higher attack detection capability. These results confirm that DTR effectively detects attacks with greater accuracy and adaptability, making it a more robust solution for IoT security.

TABLE II. PERFORMANCE COMPARISON WITH EXISTING APPROACHES USING TON-IOT DATASET

Ref.	Model	Accuracy	Precision	Recall	F-score
[7]	GADAD-ET	95	95	95	94
[8]	CNN-LSTM	98.75	98.75	98.75	98.75
[10]	DEEPShield-Ensemble	99	99	99	99
	DEEPShield-Pruning-Ensemble	99	99	99	99
[11]	LSTM-ResNet	99.24	99.18	99.15	99.16
[12]	Bi-GRU-LSTM	99.55	99.31	99.24	99.39
[13]	GLDAE	97.67	98.21	97.67	97.84
Proposed	**DTR**	**99.9284**	**99.8844**	**99.9742**	**99.8793**

IV. CONCLUSION

The IoT has revolutionized modern technology by enabling seamless communication between smart devices, edge networks, and cloud infrastructure. However, the security of IoT-Edge-Cloud networks remains a critical challenge, as these devices are vulnerable to various cyberattacks, including link attacks. Existing ML and DL approaches have been widely used for attack detection, yet fail to maintain high accuracy when attack patterns change over time. This study addresses this issue by proposing the DTR model for attack prediction and adaptation in an IoT-Edge-Cloud environment. The DTR model was implemented using SENSORIA for IoT data collection and CloudSim for edge-cloud simulation, ensuring efficient attack prediction. A statistical monitoring approach was integrated to dynamically detect shifts in attack patterns. The model was evaluated on two datasets, ToN-IoT and UNSW-NB15, demonstrating outstanding performance. On the UNSW-NB15 dataset, the model achieved 99.96% accuracy, and on the ToN-IoT dataset, it attained 99.92% accuracy. The findings show that the DTR model outperforms existing approaches, particularly in detecting evolving attack patterns. This study establishes a highly effective and adaptive attack detection model for securing IoT-Edge-Cloud networks. The DTR model not only improves attack detection accuracy but also dynamically adapts to changes in attack behavior, ensuring robust IoT security. In the future, an ensemble model combining ML/DL techniques will be developed and tested on diverse datasets to further enhance detection accuracy and adaptability in IoT security.

REFERENCES

[1] M. E. E. Alahi *et al.*, "Integration of IoT-Enabled Technologies and Artificial Intelligence (AI) for Smart City Scenario: Recent Advancements and Future Trends," *Sensors*, vol. 23, no. 11, May 2023, Art. no. 5206, https://doi.org/10.3390/s23115206.

[2] F. C. Andriulo, M. Fiore, M. Mongiello, E. Traversa, and V. Zizzo, "Edge Computing and Cloud Computing for Internet of Things: A Review," *Informatics*, vol. 11, no. 4, Sep. 2024, Art. no. 71, https://doi.org/10.3390/informatics11040071.

[3] G. I. Arcas, T. Cioara, I. Anghel, D. Lazea, and A. Hangan, "Edge Offloading in Smart Grid," *Smart Cities*, vol. 7, no. 1, pp. 680–711, Feb. 2024, https://doi.org/10.3390/smartcities7010028.

[4] S. K. Sahu and K. Mazumdar, "Exploring security threats and solutions Techniques for Internet of Things (IoT): from vulnerabilities to vigilance," *Frontiers in Artificial Intelligence*, vol. 7, May 2024, Art. no. 1397480, https://doi.org/10.3389/frai.2024.1397480.

[5] N. Singh, R. Buyya, and H. Kim, "Securing Cloud-Based Internet of Things: Challenges and Mitigations," *Sensors*, vol. 25, no. 1, Dec. 2024, Art. no. 79, https://doi.org/10.3390/s25010079.

[6] P. Mahadevappa, R. Al-amri, G. Alkawsi, A. Alkahtani, M. Alghenaim, and M. Alsamman, "Analyzing Threats and Attacks in Edge Data Analytics within IoT Environments," *IoT*, vol. 5, no. 1, pp. 123–154, Mar. 2024, https://doi.org/10.3390/iot5010007.

[7] M. F. Saiyed and I. Al-Anbagi, "A Genetic Algorithm- and t-Test-Based System for DDoS Attack Detection in IoT Networks," *IEEE Access*, vol. 12, pp. 25623–25641, 2024, https://doi.org/10.1109/ACCESS.2024.3367357.

[8] S. Yaras and M. Dener, "IoT-Based Intrusion Detection System Using New Hybrid Deep Learning Algorithm," *Electronics*, vol. 13, no. 6, Mar. 2024, Art. no. 1053, https://doi.org/10.3390/electronics13061053.

[9] I. Priyadarshini, "Anomaly Detection of IoT Cyberattacks in Smart Cities Using Federated Learning and Split Learning," *Big Data and Cognitive Computing*, vol. 8, no. 3, Feb. 2024, Art. no. 21, https://doi.org/10.3390/bdcc8030021.

[10] M. F. Saiyed and I. Al-Anbagi, "Deep Ensemble Learning With Pruning for DDoS Attack Detection in IoT Networks," *IEEE Transactions on Machine Learning in Communications and Networking*, vol. 2, pp. 596–616, 2024, https://doi.org/10.1109/TMLCN.2024.3395419.

[11] B. Cui, Y. Chai, Z. Yang, and K. Li, "Intrusion Detection in IoT Using Deep Residual Networks with Attention Mechanisms," *Future Internet*, vol. 16, no. 7, Jul. 2024, Art. no. 255, https://doi.org/10.3390/fi16070255.

[12] D. Javeed, T. Gao, M. S. Saeed, and P. Kumar, "An Intrusion Detection System for Edge-Envisioned Smart Agriculture in Extreme Environment," *IEEE Internet of Things Journal*, vol. 11, no. 16, pp. 26866–26876, Aug. 2024, https://doi.org/10.1109/JIOT.2023.3288544.

[13] C. Yang, L. Wu, J. Xu, Y. Ren, B. Tian, and Z. Wei, "Graph Learning Framework for Data Link Anomaly Detection," *IEEE Access*, vol. 12, pp. 114820–114828, 2024, https://doi.org/10.1109/ACCESS.2024.3445533.

[14] J. F. Kurian and M. Allali, "Detecting drifts in data streams using Kullback-Leibler (KL) divergence measure for data engineering applications," *Journal of Data, Information and Management*, vol. 6, no. 3, pp. 207–216, Sep. 2024, https://doi.org/10.1007/s42488-024-00119-y.

[15] J. N. Al-Karaki and G. A. Al-Mashaqbeh, "SENSORIA: A New Simulation Platform for Wireless Sensor Networks," in *2007 International Conference on Sensor Technologies and Applications (SENSORCOMM 2007)*, Valencia, Spain, Oct. 2007, pp. 424–429, https://doi.org/10.1109/SENSORCOMM.2007.4394958.

[16] T. Goyal, A. Singh, and A. Agrawal, "Cloudsim: simulator for cloud computing infrastructure and modeling," *Procedia Engineering*, vol. 38, pp. 3566–3572, 2012, https://doi.org/10.1016/j.proeng.2012.06.412.

[17] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, Nov. 2015, pp. 1–6, https://doi.org/10.1109/MilCIS.2015.7348942.

[18] N. Moustafa and J. and Slay, "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," *Information Security Journal: A Global Perspective*, vol. 25, no. 1–3, pp. 18–31, Apr. 2016, https://doi.org/10.1080/19393555.2015.1125974.

[19] N. Moustafa, J. Slay, and G. Creech, "Novel Geometric Area Analysis Technique for Anomaly Detection Using Trapezoidal Area Estimation on Large-Scale Networks," *IEEE Transactions on Big Data*, vol. 5, no. 4, pp. 481–494, Sep. 2019, https://doi.org/10.1109/TBDATA.2017.2715166.

[20] N. Moustafa, G. Creech, and J. Slay, "Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models," in *Data Analytics and Decision Support for Cybersecurity: Trends, Methodologies and Applications*, I. Palomares Carrascosa, H. K. Kalutarage, and Y. Huang, Eds. Springer International Publishing, 2017, pp. 127–156.

[21] M. Sarhan, S. Layeghy, N. Moustafa, and M. Portmann, "NetFlow Datasets for Machine Learning-Based Network Intrusion Detection Systems," in *Big Data Technologies and Applications*, 2021, pp. 117–135, https://doi.org/10.1007/978-3-030-72802-1_9.

[22] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustainable Cities and Society*, vol. 72, Sep. 2021, Art. no. 102994, https://doi.org/10.1016/j.scs.2021.102994.

[23] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. H. den Hartog, "ToN_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, Jan. 2022, https://doi.org/10.1109/JIOT.2021.3085194.

[24] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, "TON_IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 165130–165150, 2020, https://doi.org/10.1109/ACCESS.2020.3022862.

[25] N. Moustafa, M. Keshky, E. Debiez, and H. Janicke, "Federated TON_IoT Windows Datasets for Evaluating AI-Based Security Applications," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Dec. 2020, pp. 848–855, Guangzou, China, https://doi.org/10.1109/TrustCom50675.2020.00114.

[26] N. Moustafa, M. Ahmed, and S. Ahmed, "Data Analytics-Enabled Intrusion Detection: Evaluations of ToN_IoT Linux Datasets," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Guangzhou, China, Dec. 2020, pp. 727–735, https://doi.org/10.1109/TrustCom50675.2020.00100.

[27] N. Moustafa, "A Systemic IoT-Fog-Cloud Architecture for Big-Data Analytics and Cyber Security Systems: A Review of Fog Computing." arXiv, May 04, 2019, https://doi.org/10.48550/arXiv.1906.01055.

[28] J. Ashraf *et al.*, "IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities," *Sustainable Cities and Society*, vol. 72, Sep. 2021, Art. no. 103041, https://doi.org/10.1016/j.scs.2021.103041.

[29] N. Moustafa, "New Generations of Internet of Things Datasets for Cybersecurity Applications based Machine Learning: TON_IoT Datasets," presented at the eResearch Australasia Conference, Brisbane, Australia, Oct. 2019.