Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X
Volume 4, Issue 1, 2026

AN ADAPTIVE ZERO-DAY INTRUSION DETECTION AND TRAFFIC CLASSIFICATION FRAMEWORK FOR IOT NETWORKS USING AUTOENCODER-BASED ANOMALY ISOLATION

Sania Sajid[*1], Jawaid Iqbal[2], Azeem Akram[3]

[*1]*Master of Science in Computer Science, from Riphah International University, Islamabad*
[2]*Associate Professor, Faculty of Computing, Riphah International University, Islamabad*
[3]*Master of Science in Software Engineering, Riphah International University, Islamabad*

[*1]saniasajid115@gmail.com , [2]Jawaid.iqbal@riphah.edu.pk , [3]akramazeem947@gmail.com

DOI: https://doi.org/10.5281/zenodo.18277039

Keywords
IoT Security; Zero-Day Attacks; Intrusion Detection System; Autoencoder; Adaptive Thresholding; Anomaly Detection

Article History
Received: 25 November 2025
Accepted: 01 January 2025
Published: 16 January 2026

Copyright *@Author*

Corresponding Author: *
Sania Sajid

Abstract

The rapid expansion of Internet of Things (IoT) networks has significantly increased security risks due to heterogeneous device behavior, constrained computational resources, and the continuous emergence of novel cyberattacks. Traditional intrusion detection systems (IDS) predominantly rely on predefined attack signatures or supervised learning approaches that require labeled attack data, making them ineffective against previously unseen zero-day attacks. This paper presents a deployment-oriented adaptive intrusion detection framework for zero-day attack detection in IoT networks using autoencoder-based anomaly isolation. The proposed autoencoder is trained exclusively on benign IoT traffic to learn normal behavioral patterns without relying on attack signatures. Anomaly detection is initially performed using static thresholding and subsequently enhanced through an adaptive thresholding mechanism that dynamically adjusts the decision boundary based on recent traffic statistics. To evaluate real-world applicability, live IoT traffic is captured using Wireshark, transformed into flow-based features using CICFlowMeter, and analyzed through an online adaptive detection process without retraining the model. Experimental results show that static thresholding performs poorly under dynamic and imbalanced traffic conditions, achieving only 36% detection accuracy. In contrast, the proposed adaptive thresholding approach achieves detection accuracy of up to 96–97% on benchmark 18 datasets while effectively reducing false alarms. Validation on real IoT traffic reveals that 19 approximately 10% of flows are identified as anomalous, reflecting realistic deployment 20 behavior rather than excessive false positives. The lightweight nature of the proposed model, with a memory footprint below 0.5 MB and an inference latency of approximately 62 ms, demonstrates its suitability for real-time IoT deployments.

1.Introduction

The Internet of Things (IoT) has enabled large-scale deployment of interconnected devices across diverse application domains such as healthcare, smart homes, industrial automation, and smart cities. Despite their rapid adoption, IoT networks remain highly vulnerable to cyber threats due to limited device resources, heterogeneous communication protocols, weak authentication mechanisms, and continuous exposure to open network environments., Traditional intrusion detection systems (IDS) predominantly rely on signature-based techniques or supervised learning models. Signature-based approaches are ineffective against previously unseen attacks, while supervised learning methods require labeled attack data that is often unavailable or incomplete in real-world IoT deployments. As a result, zero-day attacks continue to pose a significant challenge to IoT security infrastructures. To overcome these limitations, recent research has explored anomaly-based intrusion detection approaches using machine learning and deep learning techniques. In particular, autoencoder-based models have gained attention due to their ability to learn normal traffic behavior in an unsupervised manner. However, many existing approaches rely on static decision thresholds, offline evaluation, or synthetic datasets, which significantly limits their effectiveness in dynamic and heterogeneous IoT environments. Moreover, the majority of existing studies lack validation using real IoT traffic captured from operational networks. In this context, this paper presents an adaptive intrusion detection framework for IoT networks that focuses on practical deployment and zero-day attack detection. The proposed framework employs an autoencoder trained exclusively on benign IoT traffic to model normal network behavior without relying on attack signatures or labeled data. An adaptive thresholding mechanism is introduced to dynamically adjust the anomaly detection boundary based on recent traffic statistics, thereby reducing false alarms under evolving network conditions. In addition, the framework is validated using real IoT traffic captured via Wireshark and processed using CICFlowMeter, enabling realistic evaluation without retraining the detection model. Furthermore, post-detection attack classification is performed to enhance interpretability while preserving the integrity of zero-day detection. The remainder of this paper is organized as follows. Section 2 reviews related work on IoT intrusion detection. Section 3 describes the proposed methodology and system architecture. Section 4 presents the experimental results and performance evaluation. Section 5 discusses the findings, and Section 6 concludes the paper with directions for future research.

2. Related Work

The detection of zero-day attacks in IoT networks has attracted significant research interest due to the scarcity of labeled attack data and the continuously evolving threat landscape. To address these challenges, researchers have increasingly explored anomaly-based intrusion detection approaches using machine learning and deep learning techniques. Meidan et al. [1] proposed a machine learning-based framework to model normal IoT network behavior for detecting unauthorized devices. Their approach demonstrated that learning benign traffic patterns can be effective for anomaly detection; however, the detection process relied on static decision boundaries and was evaluated primarily on controlled datasets, limiting its robustness in dynamic environments. Ferrag et al. [2] provided a comprehensive survey of deep learning-based intrusion detection systems for IoT networks. The study highlighted that although many proposed models achieve high accuracy on benchmark datasets, most of them lack real-world deployment validation and adaptive mechanisms to handle traffic variability.Hindy et al. [3] investigated deep learning techniques for detecting advanced and zero-day network threats and emphasized the importance of anomaly-based detection. Nevertheless, their work remained largely conceptual and did not include experimental validation using real IoT traffic captured from operational environments. Abdulhammed et al. [4] focused on supervised machine learning approaches combined with feature selection techniques for IoT intrusion detection. While their framework achieved strong performance for known attacks, its reliance on labeled attack data inherently limits its applicability for zero-day attack detection. Almiani et al. [5] proposed an unsupervised deep learning-based intrusion detection system for IoT networks. Although effective on benchmark datasets, the proposed method relied on static thresholding and offline evaluation, which may lead to elevated false alarm rates when deployed in dynamic real-world IoT environments. Overall, existing studies demonstrate the potential of unsupervised and deep learning- based intrusion detection for IoT networks; however, most approaches rely on static thresholds, offline datasets, or supervised learning assumptions. Moreover, limited work validates intrusion detection frameworks using real IoT traffic or integrates adaptive detection mechanisms capable of adjusting to evolving traffic behavior. These limitations motivate the need for deployment-oriented intrusion detection frameworks that combine benign-only training, adaptive thresholding, and real-world validation.

3. Research Questions

This study is guided by the following research questions:

- RQ1: Can an autoencoder trained exclusively on benign IoT traffic effectively detect zero-day attacks?
- RQ2: Why does static thresholding lead to high false alarm rates, and how does adaptive thresholding improve detection reliability?
- RQ3: Does the proposed frame work generalize effectively to real IoT traffic captured from an operational environment?
- RQ4: Can post-detection attack classification be performed without compromising zero-day detection capability?

The proposed research questions motivate the design of a complete detection pipeline that captures the practical constraints of real-world IoT environments. To this end, the following section presents the architecture and operational flow of the proposed intrusion detection framework, highlighting how benign-only training, adaptive thresholding, and real traffic validation are integrated into a unified detection process. In addition, the framework is designed to operate without reliance on prior attack knowledge, ensuring robustness against emerging and unknown threats. The emphasis on deployment feasibility distinguishes the proposed approach from purely offline or simulation-based detection models.

4. Materials and Methods

This section describes the overall detection pipeline, system architecture, model training strategy, adaptive thresholding mechanism, and datasets used to evaluate the proposed framework. The methodological design prioritizes practical deployment considerations, including computational efficiency, adaptability to traffic variations, and compatibility with real IoT network environments. Each component is structured to support real-time intrusion detection without requiring frequent retraining.

4.1. System Architecture

The overall architecture of the proposed adaptive intrusion detection framework is illustrated in Figure 1. The framework follows a deployment-oriented pipeline consisting of traffic capture, feature extraction, anomaly detection, adaptive thresholding, and post-detection classification. This modular architecture enables seamless integration with operational IoT networks and supports continuous monitoring of network traffic. Moreover, the separation of detection and classification stages ensures that zero-day detection capability is preserved while enhancing interpretability.

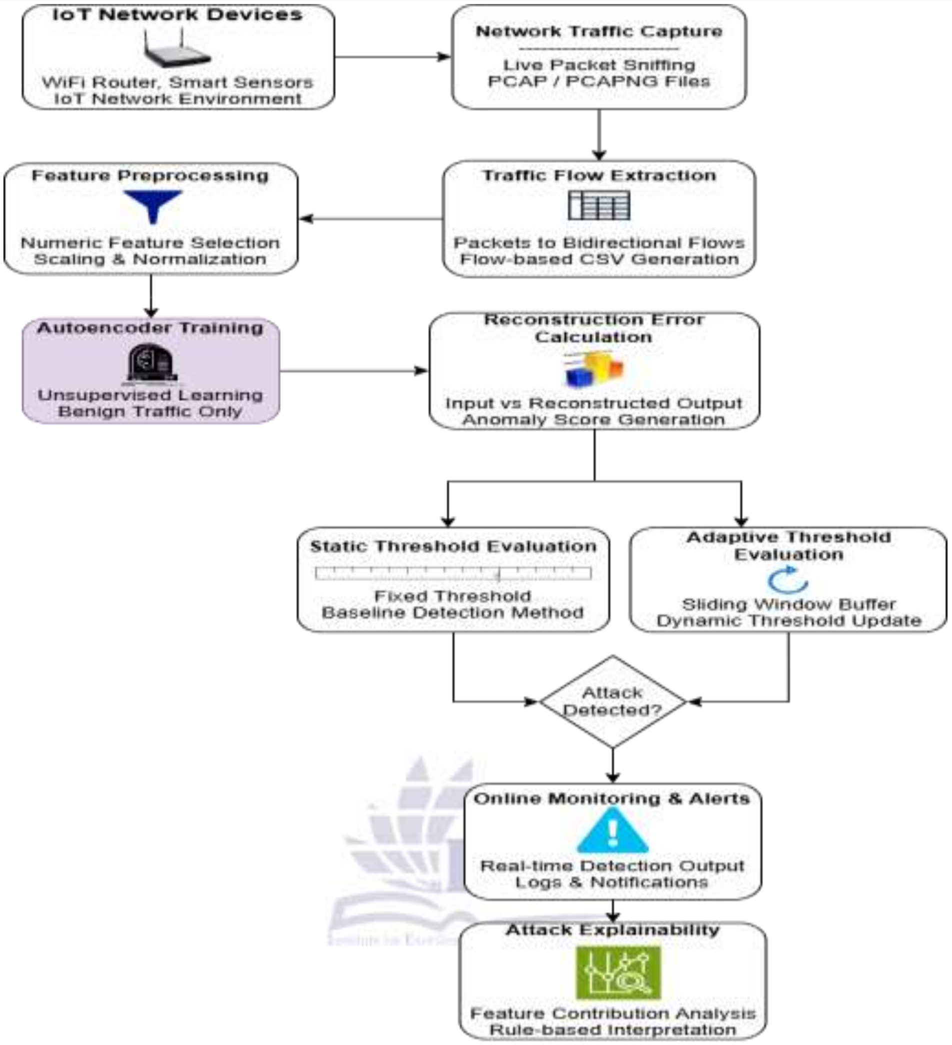


Figure 1 Work Flow

Raw IoT traffic is captured either from benchmark datasets or live network environments using Wireshark. Packet-level traffic is transformed into flow-based statistical features using CICFlowMeter. These features are normalized and provided as input to the autoencoder model, which computes reconstruction errors for anomaly detection.

4.2. Training Phase:

Learning Normal IoT Behavior The autoencoder is trained exclusively on benign IoT traffic to learn a compact representation of normal network behavior. No attack samples are included during training in order to preserve zero-day detection capability. The training objective is to minimize the reconstruction error between the input feature vector and the reconstructed output produced by the autoencoder. Mean Squared Error (MSE) is used as the reconstruction loss function. By learning only benign patterns, the model produces higher reconstruction errors when exposed to anomalous or previously unseen attack traffic.

4.3. Testing Phase:

Benchmark Dataset Evaluation After training, the autoencoder is evaluated on benchmark datasets containing both benign and attack traffic. For each flow, the reconstruction error is computed and compared against a predefined threshold to determine whether the flow is classified as normal or anomalous. Initially, a static threshold is applied to assess baseline detection performance and highlight the limitations of fixed decision boundaries in dynamic IoT environments.

4.4. Static Thresholding

The static threshold is derived from the distribution of reconstruction errors obtained from benign training data. Specifically, the threshold is set at the 99th percentile of benign reconstruction errors. Any flow with a reconstruction error exceeding this threshold is classified as anomalous. Although simple to implement, static thresholding assumes stationary traffic behavior and fails to adapt to evolving network conditions, often resulting in a high false alarm rate in real-world deployments.

4.5. Adaptive Thresholding Mechanism

To overcome the limitations of static thresholding, an adaptive thresholding mechanism is introduced. The adaptive threshold dynamically adjusts the decision boundary 147 based on recent reconstruction error statistics observed during online operation. A sliding window is maintained to store reconstruction errors corresponding only to flows classified as normal. The adaptive threshold at time t is computed as:

$$T_{adaptive}(t) = \mu t + 3\sigma t \qquad (1)$$

where μt and σt denote the mean and standard deviation of reconstruction errors within the current sliding window, respectively.

4.5.1. Sliding Window Update Rule

The sliding window is updated according to the following rule

$$W_{t+1} = \begin{cases} W_t \cup \{E_t\}, & \text{if } E_t \leq T_{\text{adaptive}}(t) \\ W_t, & \text{otherwise} \end{cases}$$

(2)

where Et represents the reconstruction error of the current flow. Only flows classified as normal are allowed to update the sliding window to prevent contamination of the adaptive thresholds by malicious traffic. A short warm-up phase is applied before adaptation begins to ensure stable threshold estimation. Since the mechanism does not rely on attack labels or retraining, it preserves the zero-day detection capability of the framework.

4.6. Real IoT Traffic Validation

To validate real-world applicability, live IoT traffic is captured from an operational network environment using Wireshark. The captured packet-level traffic is converted into flow-based representations using CICFlowMeter, as shown in Figure 2

A brief benign calibration phase is applied to align feature scaling and initial threshold values using unlabeled real traffic. No attack samples or labels are used during this phase, ensuring that zero-day detection capability is preserved.

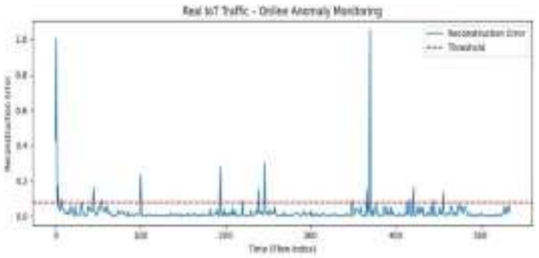


Figure 2 Real IoT traffic capture and processing using Wireshark and CICFlowMeter.

5. Dataset

5.1. Benchmark Dataset

Experiments are conducted using the CIC-IoT 2024 dataset, which contains realistic benign and malicious traffic representing modern IoT environments. Only benign samples are used during training, while attack samples are reserved exclusively for testing and evaluations.

5.2. Real IoT Traffic Dataset

In addition to benchmark datasets, real IoT traffic is captured using Wireshark from an operational environment. The resulting dataset contains unlabeled traffic flows and reflects realistic deployment conditions.

5.3. Feature Extraction

Using CICFlowMeter Packet capture (PCAP) files are processed using CICFlowMeter to extract flow-based statistical features. The extracted features include flow duration, packet counts, byte statistics, inter-arrival times, and header-level attributes. More than 80 numerical features are generated per flow.

5.4. Data Preprocessing

Non-numerical attributes are removed, and missing values are replaced with zeros. Feature scaling is performed using standard normalization to ensure consistent input 187 distributions. The same preprocessing pipeline is applied across benchmark datasets and 188 real IoT traffic to avoid data leakage.

6. Results

This section presents the experimental evaluation of the proposed adaptive intrusion detection framework using benchmark datasets and real IoT traffic.

6.1. Online Adaptive Detection on Real IoT Traffic

Figure 3 illustrates the reconstruction error observed during online analysis of real IoT traffic along with the dynamically adjusted adaptive threshold.

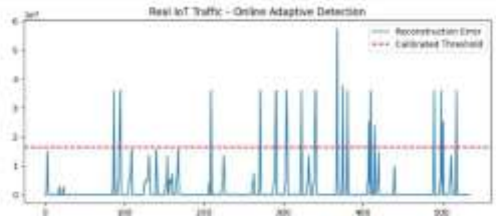


Figure 3 Online adaptive anomaly detection on real IoT traffic showing reconstruction error and calibrated threshold

The results indicate that the adaptive threshold successfully tracks gradual variations in benign traffic behavior while maintaining sensitivity to anomalous deviations.

6.2. Reconstruction Error Distribution

Figure 4 shows the reconstruction error distributions for benign traffic, attack traffic, and real IoT traffic.

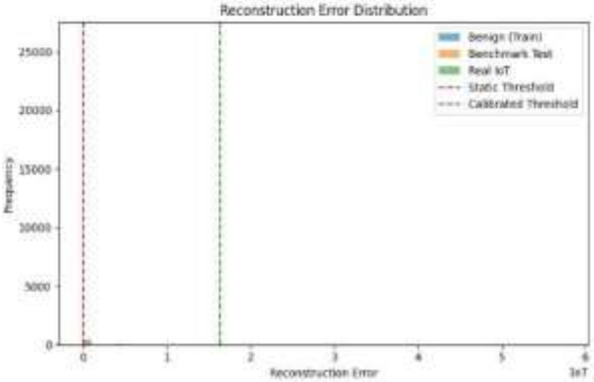


Figure 4 Reconstruction Error Distribution

Attack traffic exhibits significantly higher reconstruction errors compared to benign flows, validating the effectiveness of benign-only training for anomaly isolation.

6.3. Anomaly Ratio in Real IoT Traffic

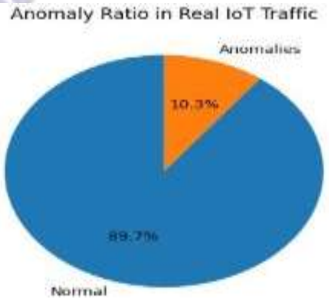


Figure 5 Anomaly Ration in Real IOT Traffic

Approximately 10% of real IoT flows are identified as anomalous, reflecting realistic deployment behavior rather than excessive false alarms.

6.4. Confusion Matrix Analysis

Figures 6 and 7 compare detection performance using static and adaptive thresholding.

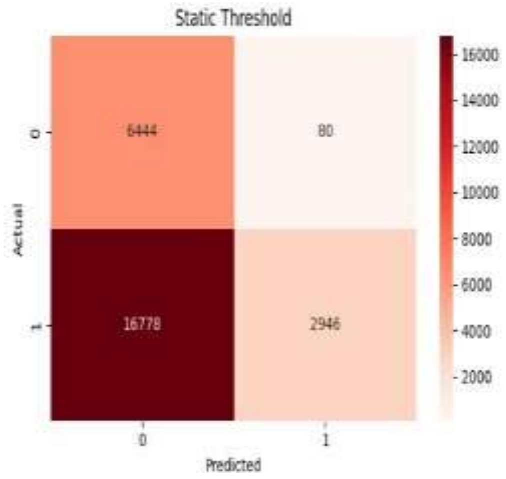


Figure 6 Static Threshold

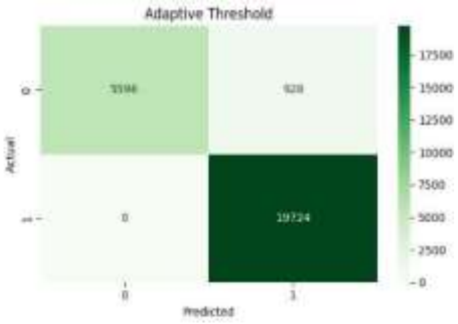


Figure 7 Adaptive Threshold

6.5. Method Comparison

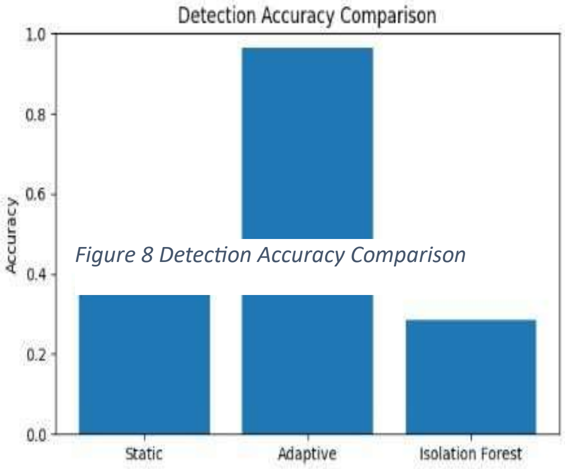


Figure 8 Detection Accuracy Comparison

Table 1 Comparison Results of Static Threshold, Isolation Forest, Proposed Adaptive IDS

6.6. Feature Contribution Analysis

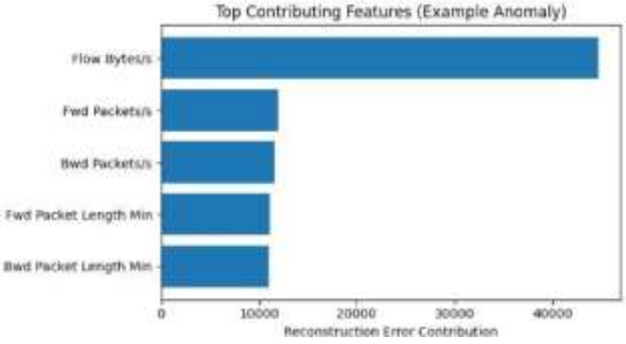


Figure 9 Feature Contribution

6.7. Post-Detection Attack Classification

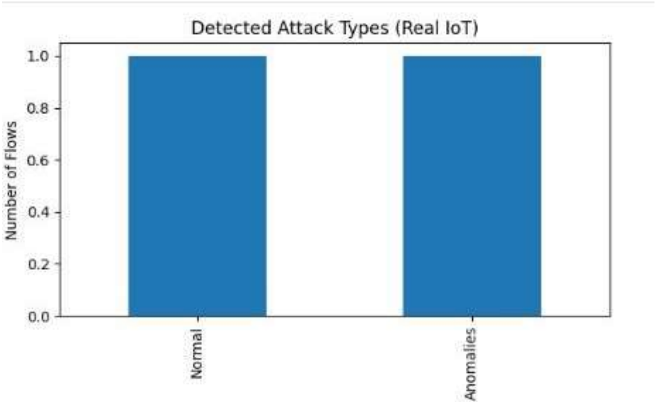


Figure 6 Detected Attack Types

7. Discussion

The results demonstrate that static thresholding is inadequate for real-world IoT intrusion detection due to the dynamic and heterogeneous nature of network traffic. Adaptive thresholding effectively adjusts detection sensitivity in response to traffic variations, resulting in a substantial reduction in false alarms. Training the autoencoder exclusively on benign traffic enables the framework to remain independent of predefined attack signatures, thereby supporting zero-day detection. Furthermore, post-detection attack classification enhances interpretability without interfering with anomaly isolation, making the framework suitable for deployment in operational IoT environments.

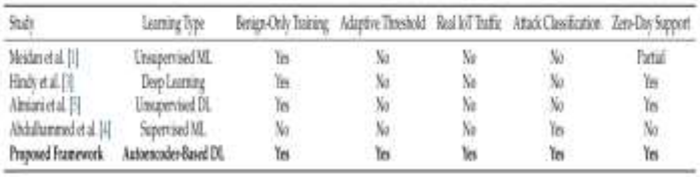
8. Answers to Research Questions

RQ1: The autoencoder trained exclusively on benign IoT traffic successfully isolates zero-day attacks by producing higher reconstruction errors for previously unseen malicious 225 traffic.

RQ2: Static thresholding fails under dynamic traffic conditions due to fixed decision boundaries. Adaptive thresholding dynamically adjusts detection sensitivity, significantly reducing false alarm. **RQ3:** The proposed framework generalizes effectively to real IoT traffic captured using Wireshark without retraining, demonstrating practical deployment feasibility.

RQ4: Post-detection attack classification improves interpretability while preserving zero-day detection capability.

Table 2 Comparison of Proposed Framework



Study	Learning Type	Benign-Only Training	Adaptive Threshold	Real IoT Traffic	Attack Classification	Zero-Day Support
Meidan et al. [1]	Unsupervised ML	Yes	No	No	No	Partial
Hindy et al. [3]	Deep Learning	Yes	No	No	No	Yes
Almiani et al. [5]	Unsupervised DL	Yes	No	No	No	Yes
Abdulhammed et al. [4]	Supervised ML	No	No	No	Yes	No
Proposed Framework	**Autoencoder-Based DL**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

9.Conclusions

This paper presented an adaptive intrusion detection framework for IoT networks that integrates benign only autoencoder training, adaptive thresholding, real IoT traffic validation, and post-detection attack classification. Experimental results demonstrate high detection accuracy with low false alarm rates, confirming the effectiveness of the proposed approach for zero-day attack detection in dynamic IoT environments.

10.FutureWork

Future research will focus on deploying the proposed framework on resource constrained edge devices, incorporating online learning mechanisms, and extending attack classification to finer grained categories.

11.References

[1] Y. Meidan, A. Shabtai, L. Rokach, Y. Elovici, and C. Glezer, “Detection of unauthorized IoT devices using machine learning techniques,” IEEE Internet of Things Journal, vol. 9, no. 5, pp. 3493–3506, 2022, doi: 10.1109/JIOT.2021.3099864.

[2] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, “Deep learning-based intrusion detection for IoT networks: A survey,” IEEE Internet of Things Journal, vol. 9, no. 6, pp. 4567–4592, 2022, doi: 10.1109/JIOT.2021.3113569.

[3] H. Hindy, E. Bayne, M. Bures, R. Atkinson, C. Tachtatzis, and X. Bellekens, “A taxonomy of network threats and the effect of deep learning on intrusion detection,” IEEE Communications Surveys & Tutorials, vol. 25, no. 1, pp. 222–259, 2023, doi: 10.1109/COMST.2022.3201457.

[4] R. Abdulhammed, A. Musa, A. A. Al-Haija, and A. Al-Shargabi, “Feature selection and machine learning-based intrusion detection for IoT networks,” Future Generation Computer Systems, vol. 139, pp. 33–45, 2023, doi: 10.1016/j.future.2022.09.028.

[5] M. Almiani, A. Abu-Shanab, and M. Al-Zoubi, “Unsupervised deep learning-based intrusion detection system for IoT networks,” Sensors, vol. 24, no. 3, Art. no. 1025, 2024, doi: 10.3390/s24031025.

[6] M. A. Alsuwaiket, “ZeroDay-LLM: A large language model framework for zero-day threat detection,” Information, vol. 16, no. 2, Art. no. 115, 2025, doi: 10.3390/info16020115.

[7] A. Mirza, S. Arshad, M. Ali, and K. Mahmood, “ZDBERTa: Advancing zero-day cyberattack detection in Internet of Vehicles with zero-shot learning,” Computers, vol. 14, no. 1, Art. no. 12, 2025, doi: 10.3390/computers14010012.

[8] A. Kumar, P. Sharma, and R. Singh, “An intelligent zero-day attack detection system using unsupervised machine learning,” Knowledge-Based Systems, vol. 285, Art. no. 111250, 2025, doi: 10.1016/j.knosys.2024.111250.

[9] A. Arizal, R. Hidayat, and M. R. Faisal, “Performance comparative study on zero-day malware detection,” International Journal of Innovative Computing, vol. 8, no. 2, pp. 45–56, 2024.

[10] S. Patel, J. Wang, and L. Chen, “Emerging AI threats in cybercrime: A review of zero-day attacks via machine learning and deep learning,” Knowledge and Information Systems, vol. 67, no. 4, pp. 1231–1265, 2025, doi: 10.1007/s10115-024-01985-7.

[11] R. Sommer and V. Paxson, “Outside the closed world: On using machine learning for network intrusion detection,” IEEE Symposium on Security and Privacy, pp. 305–316, 2010, doi: 10.1109/SP.2010.25.

[12] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, “A survey of network-based intrusion detection data sets,” Computers & Security, vol. 86, pp. 147–167, 2019, doi: 10.1016/j.cose.2019.06.005.

[13] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, and S. Venkatraman, “Deep learning approach for intelligent intrusion detection system,” IEEE Access, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[14] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, “A deep learning approach to network intrusion detection,” IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41–50, 2018, doi: 10.1109/TETCI.2017.2772792.

[15] Y. Yang, K. Zheng, C. Wu, Y. Yang, and X. Wang, “Network intrusion detection based on supervised and unsupervised learning,” IEEE Access, vol. 8, pp. 21230–21244, 2020, doi: 10.1109/ACCESS.2020.2969359.

[16] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, “A survey of deep learning-based network anomaly detection,” Cluster Computing, vol. *22, pp. 949–961, 2019, doi: 10.1007/s10586-017-1117-8.*

[17] M. Ahmed, A. N. Mahmood, and J. Hu, “A survey of network anomaly detection techniques,” Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016, doi: 10.1016/j.jnca.2015.11.016.

[18] 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, doi: 10.1109/COMST.2015.2494502.