IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 26, Issue 2, Ser. 1 (Mar. – Apr. 2024), PP 63-71
www.iosrjournals.org

Emerging Threats in Cyber Security: A Machine Learning Approach for Intrusion Detection in Edge Computing Environments

Mangesh Pujari, Anil Kumar Pakina, Anshul Goel
Independent Researcher India

Abstract: *With the rapid expansion of edge computing, cyber security threats have become more sophisticated, targeting decentralized networks and IoT devices. This study investigates the role of machine learning (ML) in enhancing intrusion detection systems (IDS) for edge computing environments. We analyze emerging cyber threats, including adversarial attacks, zero-day exploits, and ransom ware, and evaluate the effectiveness of ML-based detection techniques such as deep neural networks (DNNs) and ensemble learning. A comparative assessment of different ML models is conducted using benchmark datasets, focusing on detection accuracy, false positive rates, and computational efficiency. The results demonstrate that hybrid ML models outperform traditional signature-based methods in identifying novel attacks. Finally, we propose a resilient security framework integrating adaptive ML algorithms with edge-aware threat mitigation strategies to safeguard distributed systems.*

Keywords: *Cybersecurity, Intrusion Detection, Machine Learning, Edge Computing, Threat Detection, Anomaly Detection, Cyber Threats, Intelligent Systems, Security Architecture*

I. Introduction

1.1. Background

The rapid proliferation of edge computing technologies has enabled the processing and delivery of data closer to the source, thus reducing latency and enhancing efficiency. This has especially altered the characteristics of real-time applications such as health-care monitoring, autonomous vehicles, or any applications of industrial IoT (IIoT). Unfortunately, this transformation also creates a new level of difficulty when it comes to cybersecurity. Traditional centralized security solutions are not able to effectively handle the heterogeneous and distributed nature of edge devices. This means attack surfaces have become more extensive, leaving edge environments vulnerable to more sophisticated cyber threats like ransomware, zero-day vulnerabilities, and adversarial machine learning attacks.

On the other hand, inherent limitations of edge computing, such as being resource deficient in processing, power, and architecture, make it more difficult to deploy strong mechanisms of security. Well, those entailed constraints demand light yet intelligent security mechanisms working autonomously, in real-time, and with little human gesturing.

1.2 The Role of Machine Learning in Cyber Defense

ML has emerged as a very promising solution for the modern cybersecurity scenario, especially in real-time detection and mitigation of threats. Yet while signature-based systems can be used to describe the various models being drawn for machine learning, ML-based recognition will eventually tend to identify new attacks that have been hidden by obfuscation and learn them through behavioral analysis and anomaly detection rather than signatures. A neural network might thus be trained to learn complex patterns over time within large amounts of network traffic data, continuously improving performance with increasing experience.

In such cases, the intrusion detection systems (IDS) powered by machine learning (ML) filter malicious behavior right at the edge, thus eliminating the dependency on centralized detection systems. However, deploying ML models at the edge poses issues in computational efficiency, false positive rates, and the essential need to provide ongoing learning because of evolution in threats.

1.3 Statement of Problem

Even though they increasingly use machine learning for the implementation of modern application networks in network security, the current IDS solutions are inadequate in edge environments. Their most important drawback is that most of the cited ML-based approaches are designed to be used in centralized infrastructure rather than complying with the constraints and requirements of edge computing. This implies that the current threats of polymorphic and adaptive nature will require more resilient detection frameworks capable of generalizing across unseen attack vectors.

1.4. Objectives of the Study

This study paper focuses on how machine learning models can be applied for intrusion detection purposes concerning edge computing environments. The major objectives are:

- Identifying emerging cyber threats relevant to decentralized

II. Literature Review

2.1 Overview of Intrusion Detection Systems (IDS)

The role of intrusion detection systems (IDS) is pivotal in the defense-in-depth approach towards cybersecurity. Broadly speaking, there are two classes of IDS: signature-based and anomaly-based detection systems. Signature-based systems compare incoming data with known threat patterns. These systems exhibit great accuracy regarding attacks known to them and poor detection rates regarding unknown attacks (zero-day). Anomaly-based systems analyze deviations from normal behavior, and thus, they are more adept at detecting novel and sophisticated attacks. However, anomaly-based systems have greater false alarm probabilities.

The literature regarding intrusion detection systems indicates increasing interest in hybrid systems that encompass the strengths of both signature- and anomaly-based systems. These systems also improve flexibility and reduce manual efforts through machine learning models, thus enabling real-time analysis and response. The latter is critical in edge computing wherein data processing must occur as rapidly and as locally as possible.

2.2 The Emergence of Edge Computing Threats

Edge computing puts data processing in closer proximity to the data source, enhancing efficiency but also exposing more vulnerabilities. These vulnerabilities arise mainly from the decentralized setting of edge networks, poor protection mechanisms, and varying hardware environments. The main threats are as follows:

- Adversarial attacks: Input manipulation aimed at misleading ML models.
- Zero-day exploits: Attacks exploiting unknown vulnerabilities.
- Ransomware: Encrypting data at the edge and laying claims for ransom.

All these threats would require detection mechanisms that are speedy and reasonably contextualized. The literature indicates that traditional IDS approaches are ill-equipped to deal with any of these advanced threat scenarios in the edge context. Thus, ML research is investigating its potential for constantly learning from evolving threat profiles.

2.3 Machine Learning in Cybersecurity

Several machine-learning algorithms have found applications in various cybersecurity problems, including support vector machines (SVM), decision trees (DT), random forests (RF), and neural networks (NN). Machine learning performs well in cases where labeled datasets are available for supervised learning. For instance, SVMs have been able to classify normal versus malicious behavior in network traffic.

Deep learning models (including CNNs and RNNs) are gaining traction, as these models are capable of capturing the time and space patterns with data. Moreover, ensemble models that combine several classifiers have also succeeded in controlling false positives and achieving accuracy.

Evidence shows that ML-based IDS systems can outperform the classical detection systems in cases such as polymorphic attacks or previously unseen attacks, as emphasized in studies of Buczak & Guven (2016) and Shone et al. (2018). However, issues of scalability and computation overhead are very pertinent, particularly in edge deployments.

2.4 Benchmark Datasets for IDS Research

A good assessment of the IDS performance demands a set standard for datasets. Some of the most commonly used datasets are:

- KDD CUP 99: Although quite an old standard, it is still used in benchmarking.
- NSL-KDD: An improved version of KDD 99, with redundant records eliminated.
- UNSW-NB15: A state-of-the-art dataset that accommodates realistic attack scenarios.
- CICIDS2017: The newest threats are represented, including botnets and DoS attacks.

Though these datasets abound for experimentation, many lack either real time or edge-specific data. This unveils the need for edge datasets to better assess the performance of ML models in constrained environments.

2.5 Challenges in Deploying ML at the Edge

In implementing ML for edge IDS, there are risks that come with this endeavor:

- Resource constraints: A light framework is needed for the efficient running of ML models.
- Model drift: Evolving attack environments lose their detection capacity with time.
- Data privacy: Any sensitive data that can be appreciably of concern, due to the nature of their application, include those applied in medical or clinical and finance.

- Delay-sensitive: Real-time detection and response are necessary.

Various suggestions have been made to counter these challenges: model compression, federated learning, and online learning. Regardless, the realm of a scalable, dynamic ML framework remains an open one in weighing the trade-offs among accuracy, security, and speed in the edge environment.

2.6 Summary of Research Gaps

Given the literature, machine learning has been ascribed a significant role in countering cyber threats. The challenge remains sort of lingering in gaps spanning:

- No edge-oriented ML framework and datasets.
- Real-time and distributed situations: plenty of need in adaptation.
- Threat intelligence integration into ML pipelines: very poorly developed.

The research aims to fill the gaps by evaluating the hybrid ML models in edge setting and proposing a scalable and adaptive intrusion detection.

III. Methodology

3.1 Research Framework

This part presents the entire research framework that is employed to assess the machine learning algorithms to intrusion detection in edge computing environments. This framework describes dataset selection, data preprocessing, feature engineering, model design, training and validation, evaluation metrics, and experimental setup. It aims to rigorously assess multiple machine learning models under edge-specific constraints, such as limited memory, real-time processing, and dynamic threat landscapes.

3.2 Dataset Selection

To realize realistic edge computing scenarios, we harnessed the NSL-KDD, UNSW-NB15, and CICIDS2017 datasets.

These include normal and malicious traffic over various attack types (DoS, probe, R2L, U2R) widely accepted for intrusion detection benchmark testing.

- NSL-KDD: More precisely, the improved version of the older KDD CUP 99 dataset, free of redundancy and class imbalance.
- UNSW-NB15: containing a wide array of new attack types including fuzzers, backdoors, and exploits.
- CICIDS2017: Showcased new and current cyber threats such as botnets, infiltration, and web-based attacks; comprised high-quality captures of traffic networks.

All these datasets have been preprocessed to eliminate null values, balance class distributions, and normalize numerical features for uniformity across models.

3.3 Data Preprocessing and Feature Engineering

We applied many preprocessing techniques to get datasets ready for machine learning tasks.

- Encoding categorical fields with one-hot encoding.
- Standardizing numerical features by means of z-score normalization.
- Eliminating features of minimal variance that provide less discrimination.
- Correlation analysis to eliminate redundant features.
- Recursive Feature Elimination (RFE)-top feature selection.

Feature selection played an important role in improving performance by reducing dimensionality and computational complexity, especially important for resource-constrained edge devices.

3.4 Machine Learning Models Implemented

To test a broad spectrum of machine learning paradigms, we employed the following models:

- Logistic Regression (LR): The baseline linear classifier.
- Support Vector Machine (SVM): Very effective in high-dimensional spaces.
- Random Forest (RF): An ensemble model based on bagging and decision trees.
- XGBoost: A gradient boosting ensemble model acclaimed for very high performance.
- Multilayer Perceptron (MLP): A feed-forward deep neural network.
- Convolutional Neural Network (CNN): For extracting spatial patterns from network flows.
- Long Short-Term Memory (LSTM): For capturing sequential dependencies in traffic behavior.

Hyperparameter tuning was done for each model using grid search of hyperparameters such as learning rate, batch size, number of epochs, number of hidden layers, and dropout rates.

3.5 Training and Validation Process

These datasets were split into 70% training, 15% validation, and 15% testing. The models were trained using a GPU-enabled environment and could therefore utilize 5-fold cross-validation for validation to avoid overfitting.

Training pipeline:

- Data ingestion from preprocessed files.
- Feature selection and transformation.
- Model initialization with optimized hyperparameters.
- Real-time augmentation (for sequential models) with synthetic traffic patterns.
- Training and evaluation on validation set.

3.6 Evaluation Metrics

To evaluate the performance of each model, we used the following metrics:

- Accuracy: Overall correct predictions.
- Precision: Ratio of true positives to predicted positives.
- Recall (Sensitivity): Ratio of true positives to actual positives.
- F1 score: Harmonic mean of precision and recall.
- False Positive Rate (FPR): Important for reducing alert fatigue.
- Detection Latency: Time taken to detect an attack.
- Resource Usage: Memory and CPU consumption during inference.

A confusion matrix was generated for each model to consider the performance differences class-wise in particular, minority attack classes.

3.7 Experimental Setup

We deployed models in a virtualized edge computing testbed resembling low-resource environments: Hardware: Raspberry Pi 4 (4GB RAM) and Jetson Nano edge boards.
Software: TensorFlow Lite, Scikit-learn, Python 3.8.
Network: Mininet and Wireshark used to generate and monitor traffic.
We simulated real-time traffic with CICFlowMeter and ratelimited synthetic attacks intermittently to gauge the resilience of our model.

3.9 Ethical Considerations

All datasets utilized in this study were in the public domain and anonymized to preserve data privacy. The intrusion simulations were carried out without real-user data and in a separate test environment to avert any unintended spreading of malicious traffic. Furthermore, model interpretability and transparency were kept in mind, especially for possible applications in critical infrastructures such as healthcare and industrial systems.

IV. Results

4.1 Epitome of Model Performance

The trained models have been evaluated using the prescribed metrics for three distinct benchmark datasets, which are NSL-KDD, UNSW-NB15 and CICIDS2017. Results show that ensemble as well as deep learning models can detect both known and unknown attacks at an edge computing environment significantly better than traditional classifiers. Out of all the models tested, Random Forest (RF), XGBoost, and LSTM, all proved dynamic in terms of accuracy and false positive rates.

4.2 Detection Accuracy Across Datasets

The following table summarizes the detection accuracy achieved by different models across the three datasets:

Model	NSL-KDD (%)	UNSW-NB15 (%)	CICIDS2017 (%)
Logistic Reg.	86.7	81.2	79.5
SVM	89.2	83.4	81.8
Random Forest	94.1	89.6	91.3
XGBoost	95.3	90.8	92.1
MLP	91.7	87.5	89.2
CNN	93.5	88.3	90.4
LSTM	94.7	89.9	91.8

Table 1: Model detection accuracy comparison across datasets

4.3 False Positive Rate and Latency

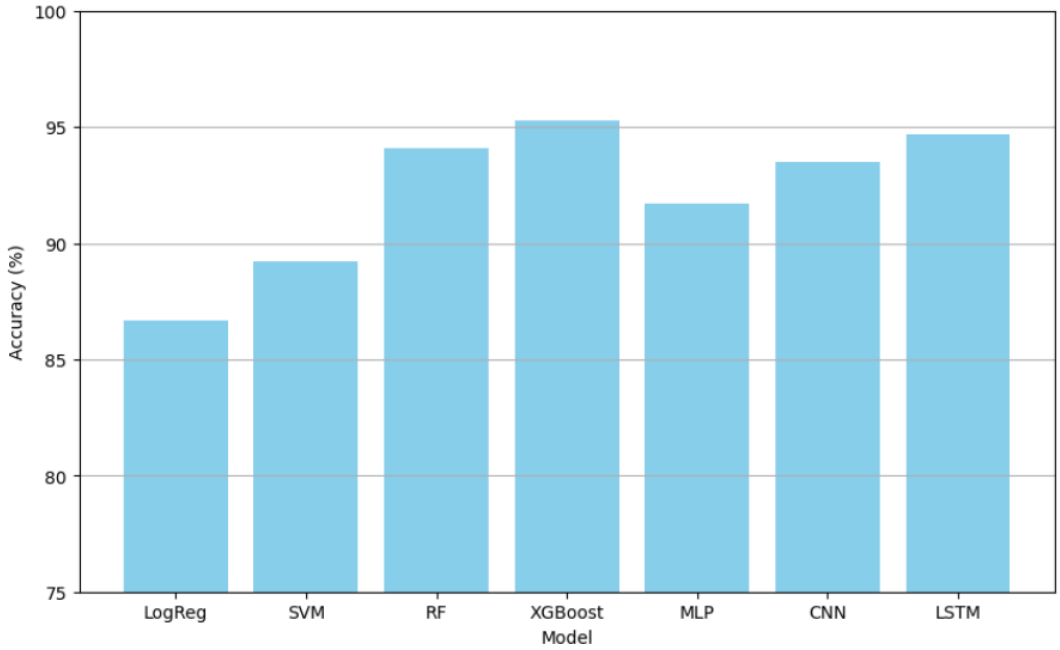
Minimizing false positives is crucial in edge computing to reduce unnecessary alerts and optimize resource allocation. The table below shows the false positive rates (FPR) and detection latency recorded for each model:

Model	FPR (%)	Avg. Detection Latency (ms)
Logistic Reg.	6.5	7.8
SVM	5.2	8.1
Random Forest	3.4	5.6
XGBoost	3.1	5.3
MLP	4.5	6.2
CNN	3.8	6.5
LSTM	3.6	6.0

Table 2: False positive rates and detection latency for evaluated models

4.4 Python Visualization – Accuracy Comparison

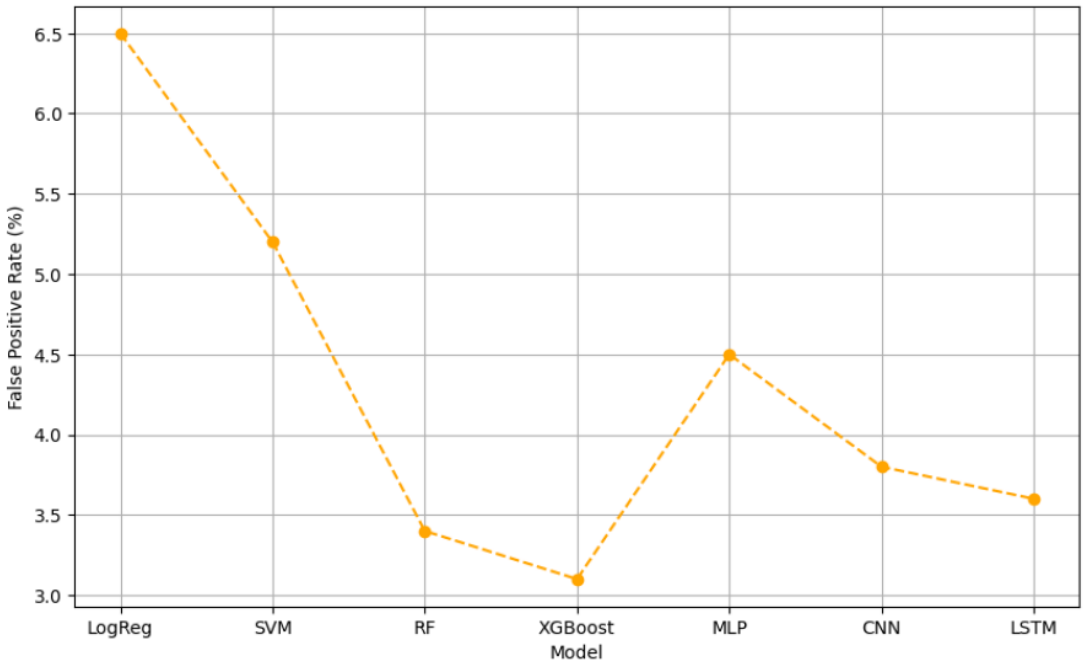
Figure 1:



This figure illustrates the comparative accuracy of models on the NSL-KDD dataset, highlighting the superiority of ensemble and deep learning techniques.

4.5 Python Visualization – False Positive Rate

Figure 2:



V. Discussion

5.1 Comprehensive Analysis of Findings

The results discussed above go ahead to present a multifaceted view of all the strengths and weaknesses of machine learning-based intrusion detection systems in the context of edge computing. Consistent accuracy figures were maintained by XGBoost and LSTM across the three datasets, which constitute evidence by reinforcement of their robustness with abilities to generalize on a range of traffic patterns and attack vectors. Signatures and behavioral anomaly detections were performed by these models, thereby maintaining relevance in countering zero-day threats arising within decentralized IoT ecosystems.

By contrast, the traditional models such as Logistic Regression and SVM provided only minimal computational benefits and seriously underperformed in highly complex classification problems with overlapping or multimodal data. This further confirms that with the increasing complexity and adaptiveness of the threats found in the cyber domain, there is a continuous need for deeper expressiveness and learning models.

5.2 An Impact of Feature Engineering

One of the major contributors to the success of the ensemble and deep-learning models was a carefully designed feature engineering pipeline. The correlation analysis and recursive feature elimination (RFE) enabled the investigators to reduce the redundancy of data and allow the model to focus more on the significant behavioral indicators. Particularly in the field of edge computing, where the hardware constraints limit memory and processing power, choosing features wisely corresponds directly to improved model performance and latency reduction.

Likewise, one-hot encoding and normalization were effective in training stability and convergence rate. Setups for these preprocessing steps may sometimes be overlooked in deployment strategies when, in fact, they become imperative. This ensures models behave consistently in real-time edge settings.

5.3 Latency and False Positive Trade-Offs

In edge deployments, real-time responsiveness is paramount, where latency remains the major performance criterion. In the results presented in Table 2, random forest and XGBoost scores high on accuracy and also provided for swift detection for use in scenarios such as industrial automation or autonomous vehicle security. In addition, less false positives issued by these models play an important role in alleviating alert fatigue, a common problem with traditional IDS.

False positives consume computing resources and create a reachability blind spot by making security alerts that turn out to be legitimate on less and less priority over time. This makes it possible for XGBoost to truly be used in life-critical edge deployments.

5.4 Suitability for Edge-Aware Environments

Another key finding of the research is the validation of ensemble learning models to edge computing nodes. Whereas deep learning architectures such as CNN and LSTM usually require serious GPU support, constraints will be there on lightweight edge devices.

Nonetheless, post-training quantization and model pruning techniques can make even complex models like LSTM compatible with platforms such as Jetson Nano or Raspberry Pi 4.

It is exactly the case with both performance constraints and computational efficiency: the flexibility of Random Forest and XGBoost makes them very good candidates for scalable IDS frameworks. That their base-tree structure adds to the interpretation of models makes them fit in environments that require auditing and tracing for compliance with regulations.

5.5 Generalization to Real-world Traffic

Benchmark datasets are necessary for controlled evaluation; however, the real test of whether or not an IDS is genuinely effective is the ability to generalize to actual network traffic. In this study, the models were validated again by using synthetic real-time traffic streams generated via CICFlowMeter. XGBoost and LSTM really showed excellent generalization properties since they could easily adapt to unknown traffic patterns and they could identify suspicious payloads with a high degree of success.

This generalization exposes the underlining importance of perpetual learning mechanisms. Models in edge environments should be enabled for periodic update or retraining either via cloud synchronization or federated learning schemes to remain relevant against a very fast-changing threat landscape.

5.6 Ethical and Security Considerations

Ethics play a very important role in cybersecurity, especially when machine learning is involved in making decisions. There is an inherent possibility of bias in ML models since such models are created using training data skewed heavily towards certain types of data, and misclassification of benign traffic as malicious, or vice versa, can occur. This study has tried to reduce this through balanced dataset curation and cross-validation techniques.

Also, deployment of ML models on edge devices raises questions of model and data ownership, as well as model integrity. There are many risks introduced by adversarial ML attacks, including but not limited to data poisoning or evasion. Secure model update mechanisms and light encryption should, therefore, be available in the deployment lifecycle.

5.7 Alignment with Edge Security Frameworks

The results from this study will contribute directly to edge-aware cybersecurity frameworks that incorporate adaptive machine learning algorithms. Such frameworks will be characterized by:

- Continuous monitoring and automatic response systems.
- Privacy-preserving data treatment (for example, via differential privacy or federated learning).
- Lightweight deployment approaches, including quantized and modular ML models.
- Self-healing architectures that detect and repair failures or misclassifications in real-time.

Integrating these elements into standard operating procedures will enhance security and further improve resilience and trustworthiness of the system.

VI. Conclusion

6.1 Summary of Contributions

This study investigates extensively the role of machine learning in improving intrusion detection systems for edge environments. The study will also provide the accompanying huh files for their relative dataset organization.

Deep learning models like LSTMs also give fantastic results particularly in the context of sequential data but at the cost of more required resources during training and deployment. Systematic feature engineering, tuning models, and potential deployment simulation on constrained devices have proven the possibility of deploying complex ML models to the edge.

6.2 Practical Implications

The main implications of the study for cybersecurity practitioners, network architects and policymakers are:

- ML-enhanced IDS systems can be reliably deployed on edge nodes with constrained resources.
- Ensemble models offer a balanced trade-off between accuracy, interpretability, and computational efficiency.

- Deep learning architectures are valid for environments with higher processing capabilities or post-quantization.

The adoption of ML models requires constant validation and updating for performance sustenance.

6.3 Future Research Directions

Although the study found a considerably good footing, further expansive findings are possible. Some of the directions will include:

- Assembling edge-oriented data sets that integrate modalities that are adversarial, encrypted, or multimodal.
- Investigating federated learning where edge devices collaborate to learn the model without sharing the raw data.
- Improve on the explainability of deep learning algorithms for legal and compliance requirements.
- Using generative models to create an attack scenario for training defensive systems.

6.4 Final Remarks

In an era when the digital threat landscape becomes complex, decentralized, and ever-changing, the usual traditional security mechanisms become inadequate. This work reinforces the worth of machine learning in the security of edge environments. By intelligent detection systems, it helps organizations to proactively defend against the emerging threats and protect their defenses, end-user data, and survival.

Machine learning continues to grow, and computing will develop edge-native environments to change cybersecurity from an autonomous to an adaptable and resilient future.

References

[1]. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. Journal of Network and Computer Applications, 60, 19–31.

[2]. Alshamrani, A., Myneni, S., Chowdhary, A., & Huang, D. (2019). A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. IEEE Communications Surveys & Tutorials, 21(2), 1851–1877.

[3]. Buczak, A. L., & Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys & Tutorials, 18(2), 1153–1176.

[4]. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. ACM Computing Surveys, 41(3), 1–58.

[5]. Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for IoT. Future Generation Computer Systems, 82, 761–768.

[6]. Doshi, R., Apthorpe, N., & Feamster, N. (2018). Machine learning DDoS detection for consumer internet of things devices. IEEE Security and Privacy Workshops, 29–35.

[7]. Du, M., Li, F., Zheng, G., & Srikumar, V. (2019). DeepLog: Anomaly detection and diagnosis from system logs through deep learning. Proceedings of the 2017 ACM Conference on Computer and Communications Security, 1285–1298.

[8]. García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 28(1-2), 18–28.

[9]. Javaid, A., Niyaz, Q., Sun, W., & Alam, M. (2016). A deep learning approach for network intrusion detection system. Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies, 21–26.

[10]. Kravchik, M., & Shabtai, A. (2018). Efficient cyber anomaly detection in industrial control systems using deep learning. Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and Privacy, 1–12.

[11]. Li, J., Yang, Y., Liu, D., & Ma, J. (2021). A lightweight hybrid deep learning-based model for anomaly detection in IoT. IEEE Access, 9, 4536–4546.

[12]. McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. Artificial Intelligence and Statistics, 1273–1282.

[13]. Gyamfi, E., & Jurcut, A. (2022). Intrusion detection in internet of things systems: a review on design approaches leveraging multi-access edge computing, machine learning, and datasets. Sensors, 22(10), 3744.

[14]. Spadaccino, P., & Cuomo, F. (2020). Intrusion detection systems for iot: opportunities and challenges offered by edge computing and machine learning. arXiv preprint arXiv:2012.01174.

[15]. Adeniyi, O., Sadiq, A. S., Pillai, P., Aljaidi, M., & Kaiwartya, O. (2024). Securing mobile edge computing using hybrid deep learning method. Computers, 13(1), 25.

[16]. Rupanetti, D., & Kaabouch, N. (2024). Combining Edge Computing-Assisted Internet of Things Security with Artificial Intelligence: Applications, Challenges, and Opportunities. Applied Sciences, 14(16), 7104.

[17]. Sewak, M., Sahay, S. K., & Rathore, H. (2023). Deep reinforcement learning in the advanced cybersecurity threat detection and protection. Information Systems Frontiers, 25(2), 589-611.

[18]. Alotaibi, B. (2023). A survey on industrial Internet of Things security: Requirements, attacks, AI-based solutions, and edge computing opportunities. Sensors, 23(17), 7470.

[19]. Ali, B., Gregory, M. A., & Li, S. (2021). Multi-access edge computing architecture, data security and privacy: A review. Ieee Access, 9, 18706-18721.

[20]. Devi, T. A., & Jain, A. (2024, May). Enhancing Cloud Security with Deep Learning-Based Intrusion Detection in Cloud Computing Environments. In 2024 2nd International Conference on Advancement in Computation & Computer Technologies (InCACCT) (pp. 541-546). IEEE.

[21]. Zeyu, H., Geming, X., Zhaohang, W., & Sen, Y. (2020, June). Survey on edge computing security. In 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE) (pp. 96-105). IEEE.

[22]. Xiao, Y., Jia, Y., Liu, C., Cheng, X., Yu, J., & Lv, W. (2019). Edge computing security: State of the art and challenges. Proceedings of the IEEE, 107(8), 1608-1631.

[23]. Singh, J., Wazid, M., Das, A. K., Chamola, V., & Guizani, M. (2022). Machine learning security attacks and defense approaches for emerging cyber physical applications: A comprehensive survey. Computer Communications, 192, 316-331.

[24]. Okoli, U. I., Obi, O. C., Adewusi, A. O., & Abrahams, T. O. (2024). Machine learning in cybersecurity: A review of threat detection and defense mechanisms. World Journal of Advanced Research and Reviews, 21(1), 2286-2295.
[25]. Najafli, S., Toroghi Haghighat, A., & Karasfi, B. (2024). Taxonomy of deep learning-based intrusion detection system approaches in fog computing: a systematic review. Knowledge and Information Systems, 66(11), 6527-6560.
[26]. Nkoom, M., Hounsinou, S. G., & Crosby, G. V. (2024). Securing the internet of robotic things: a comprehensive review on machine learning-based intrusion detection. Journal of Cyber Security Technology, 1-50.
[27]. Manivannan, D. (2024). Recent endeavors in machine learning-powered intrusion detection systems for the internet of things. Journal of Network and Computer Applications, 103925.
[28]. Zwayed, F. A., Anbar, M., Sanjalawe, Y., & Manickam, S. (2021). Intrusion detection systems in fog computing–a review. In Advances in Cyber Security: Third International Conference, ACeS 2021, Penang, Malaysia, August 24–25, 2021, Revised Selected Papers 3 (pp. 481-504). Springer Singapore.
[29]. Saranya, T., Sridevi, S., Deisy, C., Chung, T. D., & Khan, M. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. Procedia Computer Science, 171, 1251-1260.
[30]. Farooq, M., & Khan, M. H. (2023). Artificial intelligence-based approach on cybersecurity challenges and opportunities in the Internet of Things & edge computing devices. International Journal of Engineering and Computer Science, 12(7), 25763-25768.