Reviewing Machine Learning Algorithms for Threat Detection in Cybersecurity

Aidan Gatenbee, Mark Reith and Anthony Rose
Air Force Institute of Technology, Dayton, USA

aidan.gatenbee.1@us.af.mil
mark.reith.3@us.af.mil
anthony.rose.3@us.af.mil

Abstract: The recent increase in the prevalence and popularity of artificial intelligence in everyday life, such as chat agents like ChatGPT, Claude, and Google Gemini, has led to its incorporation in many fields, cybersecurity notwithstanding. In particular, machine learning, a subset of the artificial intelligence field, has led many researchers to investigate this promising technology as a method to better optimize cybersecurity applications for threat detection. The rise in machine learning comes alongside, and potentially is caused by, an ever-increasing volume of cyberattacks. As attackers have access to more sophisticated tools, using Artificial Intelligence for vectors such as social engineering and automation, cybersecurity specialists are also forced to turn to AI to match the increasing fervor. But are all algorithms created equal? The paper focuses on IEEE-sponsored journals and conferences, identifying keywords like threat detection and machine learning, using studies published within the last year. In doing so, this paper focuses on recent works that relate machine learning to cybersecurity, specifically threat detection models. This survey identifies the types of algorithms used in these applications and how they are implemented. Several baseline algorithms, such as support vector machines, k-nearest-neighbors, and convolutional neural networks, were featured in several works, while others compounded these baseline models into an ensemble algorithm, using various methods. This survey identifies strengths and weaknesses in machine learning threat detection by comparing various researched algorithms and implementations. In doing so, opportunities for future research become apparent, where researchers could attempt to defeat these algorithms or even exploit the algorithm itself to bypass detection. Another avenue for future research involves strengthening the algorithms in areas where they performed poorly by combining models into a new end product or layering them to address specific weaknesses.

Keywords: Cybersecurity, Machine Learning (ML), Artificial Intelligence (AI), Threat detection

1. Introduction

Traditional, rule-based cybersecurity approaches are no longer sufficient to address rapidly evolving cyber threats (S et al. 2025). These threats have swiftly increasing levels of complexity, incorporating Artificial Intelligence (AI) through means such as new zero-day exploits, AI-enhanced Advanced Persistent Threats (APTs), and ransomware. Manikandan, Reddy Onteddu, and Chilimi (2025) expand on this issue, further citing polymorphic malware as a significant cause for concern that can bypass signature-based or rules-based protections. This rise in AI-powered cyber threats has led to a push in demand for similarly capable resources for defense, to which industry research has responded by providing several works investigating the capabilities of, and proposing new AI-based tools for cybersecurity. This work reviews these publications, focusing on threat detection as a subdomain of cybersecurity. By comparing training techniques, utilized models, advantages, and limitations, this work produces a thorough survey of existing literature regarding AI-based threat detection models and provides subjects for future work in the research area.

1.1 Background

AI can be broken down into several subcategories. Most works relating AI to cybersecurity focus on Machine Learning (ML) and Deep Learning (DL). DL is further a subset of ML, where ML describes algorithms that learn from a provided set of data and improve by iterating over that data for increased performance. DL uses the same principles but introduces neural networks, a type of algorithmic model inspired by the human brain, in which various interconnected nodes pass data to one another to improve performance. The surveyed works use a variety of different models that fall into ML and DL, such as Decision Trees (DT), Random Forests (RF), k-Nearest Neighbors (kNN), Gradient Boosting Machines (GBM), and Support Vector Machines (SVM) for ML, while models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Memory Augmented Neural Networks (MANN), Generative Adversarial Networks (GAN), and Long Short Term Memory (LSTM) are DL models. Federated Learning (FL) describes how and where the training is performed, but is not itself a learning algorithm. FL is a learning method where models are partially trained on-site, then the model's improvements are shared with other devices in the learning network to further improve. FL aims to preserve privacy by keeping the data the model trains on internal, preventing unnecessary exposure.

DTs are a classification algorithm that attempts to sort data into the categories provided in the initial training by making binary decisions through various internal steps to sort all data. RFs are an ensemble of DTs, where many DTs are evaluated together to better sort data. kNN is a simple classification algorithm that evaluates which group a datapoint belongs to by looking at the nearest k datapoints, and sorts the new datapoint by choosing the majority of those k datapoints. GBMs are another DT ensemble that aims to evaluate complex relationships within datasets. SVMs are somewhat similar to kNNs as they aim to differentiate between classes using nearby datapoints, but SVMs do so by attempting to find the delimiter that maximizes the space between nodes of different classes.

CNNs are a neural network inspired by neural organization within the visual cortex. CNNs use fully interconnected nodes, and while they typically see use in visual detections, they can be used for anomaly detection in a variety of spaces. RNNs are also fully interlinked, but differentiate themselves from CNNs with an internal memory that the algorithm uses to take information from previous inputs to influence the current evaluation. RNNs are typically used in sequenced information, such as time-oriented data. MANNs also use a memory structure, but MANN memory is external rather than internal. This makes MANNs popular in language processing and reinforcement learning. GANs operate by creating synthetic data from the provided training data. It then uses a discriminator to determine which data is real versus synthetic. These GANs are great at providing additional data for other algorithms to refine themselves on. Lastly, LSTMs are an evolution of the RNN. RNNs struggle to recall long-term dependencies, where LSTMs are specifically designed to counter this loss.

F1-score and inference are the two metrics this work analyzes to gauge performance. F1-score is a zero-to-one scale metric balancing the impact of false positives and false negatives. F1-score is a suitable metric for analyzing cybersecurity models because a false positive indicates the solution would have blocked benign network traffic, reducing usability for clients on the network, and false negatives represent unblocked malicious traffic, both significant impact on a network. Figure 1 below is a graphical representation of this balance. Inference time represents the time it would take a proposed architecture to create an assessment of potentially malicious traffic, an important indicator for real-time solutions.

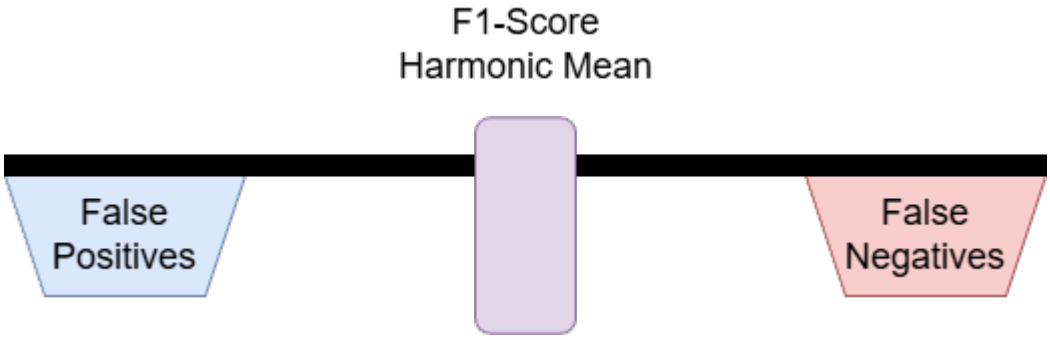

Figure 1: Graphical representation of F1-score. The weights of False Positives and False Negatives are balanced together into the F1-score, representing the harmonic mean of the two. F1-score is often preferred when False Negatives and False Positives are of equal importance

2. Literature Review

This survey reviews several works that introduce a variety of ML and DL models and techniques for cybersecurity. Modern solutions have moved towards more ensemble-based solutions rather than standalone models, and as a result have seen improved performance. Bhuktar et al. (2025) proposes an embedded cybersecurity architecture that uses a CNN-LSTM ensemble model on edge devices for low-cost, distributed computing. These learnings are combined using FL to produce a single comprehensive model. This model is then tested against synthesized data from a GAN, which aims to test system resilience “by generating adversarial scenarios and implementing countermeasures...to handle incipient threats” (Bhuktar et al. 2025). The authors report an F1-score of 98.1% and an inference time of 5.2ms.

Srilakshmi et al. (2025) also investigates Federated CNN-LSTM as an ML-based solution for threat detection. Srilakshmi et al. did not provide an F1-score, but did provide an accuracy score and false positive rate (FPR) of 98.3% and 1.2%, respectively. While the authors additionally provide a detection time metric of 160ms, they do not show their model implementation alongside this latency.

El-Hajj (2025) uses an RF-LSTM ensemble, employing RF for static features and LSTM for temporal features. The authors did not include FL, but recognize it as a target for future work improving privacy protection. El-Hajj

also includes a detailed outline of their real-time integration architecture, serving as a model example for future research to follow and expand upon. The RF-LSTM model produces an F1-score of 98.7% and an inference time of 4.2ms with 12,000 requests per second. El-Hajj goes on to showcase the specific cyber attacks their model performed best on, providing a level of depth not seen elsewhere.

S et al. (2025) proposes a GAN-augmented MANN for their "NeuroShield" model, targeting smart infrastructure and IoT networks. The GAN introduces synthetic data not seen elsewhere, aiming for improved zero-day detection and response. While S et al. provides a figure for false positive rates, it is poorly labeled and thus hard to determine their proposed model's accuracy. They provide an attack success rate, where only 6.1% of attacks intercepted by NeuroShield were successful, but without disclosing the specific attacks used, this figure has little inherent value. NeuroShield also has a response time of 310ms, but the authors do not provide the specifics of their implementation, leaving it hard to compare to other works.

Choudhary et al. (2025) use a four-way ensemble of logistic regression, DT, SVM, and kNN. The ensemble is further improved with a GBM and hyperparameter tuning. While Choudhary et al. does not go in-depth about how they construct their ensemble, they provide an F1-score of ≈91%. Choudhary et al. do not include real-time metrics in their work.

Manikandan, Reddy Onteddu, and Chilimi (2025) researched several standalone models, including SVM, DT, RF, kNN, and CNN. They report a perfect F1-score of 100% for DT and RF, raising concerns of overfitting. kNN has the next-highest F1-score of 99%, but kNN lacks the complexity to appropriately handle complex attacks. While they claim 100,000 malware and benign process sources for their dataset, this information cannot be verified without citations. While Manikandan, Reddy Onteddu, and Chilimi does not analyze a real-time implementation, they do provide relative training times. The authors found that DT and kNN had the fastest training times, claiming this makes them ideal for real-time applications. However, in deployable solutions, all training for non-adaptive models should be done prior to deployment, invalidating this claim.

C S et al. (2025) uses an RF algorithm, chosen out of DT, SVM, and kNN, for RF's ability to handle high dimensionality in the chosen dataset, CICIDS2017. This resulted in an F1-score of 97.5%. Polinati (2025) found CNN among RF, DT, SVM, and RNN to have the highest F1-score at 91.5%, testing on the NSL-KDD and CICIDS2017 datasets. Neither C S et al. nor Polinati provides real-time metrics.

There are also system-specific frameworks for cybersecurity involving ML. Edouard Bouda, Sere, and Ouedraogo (2025) provide one such example by introducing an RF model into an architecture designed to detect malicious DNS requests. Edouard Bouda, Sere, and Ouedraogo achieve this by blending several DNS protection methods together to achieve a F1-score of 99.35% and a processing time of 2.5ms. While this processing time is higher than their other components, the performance improvement justifies the additional overhead.

Other authors have also performed literature reviews in the context of ML-powered cybersecurity. Pandey et al. (2025) reviewed works from 2021 to 2024, finding several challenges in the implementation of ML-based cybersecurity. They end their work by highlighting the importance of transparency and explainability in ML solutions that can keep up with the rapid pace of evolving threats.

3. Comparative Analysis

Section 2 illustrates how Bhuktar et al. (2025) and Srilakshmi et al. (2025) both test CNN-LSTM as a solution for real-time ML-based threat detection. The works have similar figures for accuracy, but Srilakshmi et al. does not show additional statistics like F1-score, precision, or recall as in Bhuktar et al. Furthermore, Srilakshmi et al. has a much higher detection time at 160ms, whereas Bhuktar et al. boasts a 5.2ms inference time. However, direct comparisons between the two works are difficult because Srilakshmi et al. does not provide their implementation architecture, which would provide valuable insights into where the extra delay in inference/detection stems from.

As compared to Bhuktar et al. (2025) and Srilakshmi et al. (2025) using an FL-powered CNN-LSTM, El-Hajj (2025) uses an RF-LSTM model without FL, although it is noted as a topic in their future work. El-Hajj outperformed Bhuktar et al. and Srilakshmi et al. by about 0.5% in F1-score and about 1ms in inference time. Without knowing the architecture of Srilakshmi et al., time comparisons are difficult to evaluate accurately.

S et al. (2025) is difficult to contextually evaluate due to a lack of provided metrics like F1-score. S et al. reported a successful defense rate of 93.9% and a response time of 310ms, but without knowing the specifics of their attack makeup and network architecture, true comparative analysis is difficult.

The last surveyed ensemble model, Choudhary et al. (2025) reported a 91.28% F1-score and a 7.7% error rate. This significantly underperforms compared to El-Hajj (2025), Bhuktar et al. (2025), and Srilakshmi et al. (2025). This could indicate the power of LSTM for network traffic because of its highly temporal nature, as that is the common factor between El-Hajj, Bhuktar et al., and Srilakshmi et al.

Of the standalone models researched by Manikandan, Reddy Onteddu, and Chilimi (2025), C S et al. (2025), and Polinati (2025), only Manikandan, Reddy Onteddu, and Chilimi reported an F1-score higher than those of ensemble counterparts at 100% for DT and RF models. However, a perfect score raises concerns of overfitting the models to the source dataset and reducing their general usability. The next highest F1-score was C S et al. with a RF at 97.5%, still less than that of ensemble models. While Manikandan, Reddy Onteddu, and Chilimi reports a training time statistic for their researched models, this is irrelevant for real-time response, as all training should be done before deployment.

All ensemble models outperformed work done via standalone models a la Manikandan, Reddy Onteddu, and Chilimi, C S et al., and Polinati versus works done by El-Hajj, S et al., Bhuktar et al., and Choudhary et al. There is, however, an exception in Edouard Bouda, Sere, and Ouedraogo (2025) with their DNS-specific implementation of a single-model ML algorithm. For this work, however, it is important to note that the ML component is augmenting non-ML solutions and is not the whole solution in and of itself. Even still, this application may be made even stronger by implementing an ensemble model in place of a single RF. Table 1 below collates these results for easy viewing.

Table 1: Comparison between research done by surveyed literature

Author & Year	Title	Model	Metrics	Summary
Bhuktar et al., 2025	Real-Time Cyber Threats... using Embedded AI	CNN-LSTM w/ FL	98.1% F1-score, FPR 1.6%. Inf. time 5.2 ms	Embedded system for RT threat detection.
Srilakshmi et al., 2025	Real-Time IoT Cybersecurity... to Train Generative Robots	CNN-LSTM w/ FL	98.3% accuracy, FPR 1.2%. 160 ms response	Uses GenAI robots to automate security and threat responses in real-time.
El-Hajj, 2025	AI-Powered Threat Detection and Response...	RF-LSTM	98.7% F1-score. Avg latency 4.2 ms, 12,000 rqs/sec.	Hybrid framework for RT-IDS. RF-LSTM deployed with Apache Kafka and TensorRT.
S et al., 2025	Neural Cybersecurity... Deep Learning for Real-Time Threat Detection	MANN, DRL	310 ms response, firewalls 1250 ms, other AI 740 ms. 93.9\% defense, 61.3\% baseline.	Self-supervised deep learning, Cyber Swarm Intelligence, Quantum-Secure AI Encryption
Choudhary et al., 2025	AI-Driven Threat Simulation in Wireless Networks...	4-way ensemble	91.28% F1-score. and 7.7% error	Verbose ensemble w/ XGBoost, hyperparameter optimization
Manikandan et al., 2025	Next-Gen Malware Detection using AI...	DT, RF, KNN, CNN	Near-100%% acc. for DT, RF, & KNN. DT & KNN fastest, CNN 97.79% acc., costly	Feature engineering for critical information, testing on a large dataset of 100,000 samples
C S et al., 2025	AI Driven Threat Detection... using Machine Learning Techniques	RF, DT, SVM, KNN	RF 97.5% F1-score	Highly robust, effective ident. of Botnet, DDoS, and Heartbleed. Trained and tested using CIC-IDS2017.
Polinati, 2025	AI and Deep Learning-Powered Threat Intelligence	RF, DT, SVM, CNN, RNN	CNN, 93.7\% accuracy, 91.5\% F1-score. $<$2s response w/ automated response system.	Evaluates RF, DT, SVM, CNN, and RNN on datasets like NSL-KDD and CICIDS 2017.
Edouard Bouda et al., 2025	Improved Domain Name System Threat Detection	DGA + RF, RPZ + DGA + AI	DGA + RF 99.94\% accuracy, 98.81\% precision. RPZ + DGA + AI 100\% recall. 2.5ms overhead	Combines RPZ, DGA, and RF classifier for malicious DNS requests.
Pandey et al., 2025	AI-Powered Defenses: A Machine Learning Approaches...	---	---	Highlights data privacy, “black box”, vulnerable models. Hybrid models, FL for future works.Review of ML in cybersecurity. Surveys model's ability to detect various threats.

4. Challenges and Limitations

Several challenges come with pushing ML into cybersecurity. Notably, reliance on a limited number of benchmarks and integration into real-time response units. S et al. (2025) even directly mentions these weaknesses, additionally citing real-time detection and explainability as barriers to successful applications. Few current, usable benchmarks for training and developing ML tools for cybersecurity exist; Four of the cited works use CICIDS-2017, a network traffic benchmark with benign and malicious data. A further three use KDD-Cup 1999, another network traffic benchmark, while Edouard Bouda, Sere, and Ouedraogo use a DNS-specific dataset of malicious domain names. The remaining two works do not cite their datasets at all, casting doubt on their sources. The limited number of benchmarks restricts research in the area, potentially leading to stagnation. Introducing new, relevant datasets and benchmarks, or showing how the existing sources are still relevant, could help alleviate this issue.

Additionally, many ML applications are often perceived as a "black box" as described in Pandey et al. (2025), with a heavily obscured decision-making process that makes it difficult for an end-user to trust the reported outcome fully. For critical security applications, this lack of transparency makes trusting the model's decision-making, even if the model is highly accurate, difficult. This challenge cannot be easily solved within existing models. Instead, breakthroughs in explainable AI are required to handle this issue.

5. Trend and Future Directions

As shown in C S et al. (2025), El-Hajj (2025), Srilakshmi et al. (2025), Bhuktar et al. (2025), and Choudhary et al.

(2025), FL shows extreme promise for reducing the cost of training through a distributed architecture and for privacy-preserving training. This double-benefit makes FL a must-implement in all future ML models for cybersecurity, as the distributed architecture allows for continuous training without mass downtime for computationally expensive traditional training.

Given the performance gap between standalone models and ensemble models, except for novel research producing a new model, there remains little to be gained by implementing single models for ML-based cybersecurity solutions. Future works in this field should focus on how to best hybridize various models to cover individual weaknesses, and then how to implement these models into a realistic architecture to limit latency and network overhead caused by introducing the model.

Lastly, researchers should pursue explainable AI to address the explainability and trust issues described in Pandey et al. (2025). Without addressing the "black box" issue, cybersecurity specialists may never trust ML models, regardless of their performance. This trust barrier will inherently reduce the effectiveness of any solution and place additional strain on those cybersecurity specialists.

6. Conclusion

This paper surveyed recent literature on implementing ML algorithms in cybersecurity, particularly studying threat detection. Through this survey, it compared recent publications within the last year. Each researched work was measured against the others, comparing models, training techniques, and performance metrics. Ensemble models generally outperform standalone ML models, highlighting how combining different models together helps to overcome their individual weaknesses. In particular, the DL model LSTM showed high promise for analyzing network traffic because of its affinity for temporal data. FL also emerged as a critical training method, offering many benefits in preserving data privacy and enabling continuous training through a distributed architecture.

This work also identified various challenges in this field, such as a limited number of dated benchmarks and the opaque nature of the decision-making process in many ML applications. This "black box" decision-making makes it difficult to trust these applications and may hinder deployment. For future research, works should prioritize model hybridization to create new ensembles, testing different combinations to determine their effectiveness. Research should also focus on real-time architectures to limit latency and overhead while also supporting a rapid response time. Lastly, improving the transparency and explainability of these models is paramount for ensuring a smooth integration into deployable environments.

Disclaimer: The views expressed are those of the author and do not reflect the official policy or position of the US Air Force, Department of Defense, or the US Government.

Ethics Declaration: This work required no ethical clearance.

AI Declaration: This work did not use any AI tools in its creation.

References

Bhuktar, Deepak Madhukar et al. (July 2025). "Real-Time Cyber Threats and Unauthorized Access Detection Using Embedded AI". In: *2025 International Conference on Computing Technologies & Data Communication (ICCTDC)*. 2025 International Conference on Computing Technologies & Data Communication (ICCTDC), pp. 1–5. DOI: 10.1109/ICCTDC64446.2025.11158799. URL: https://ieeexplore.ieee.org/document/11158799/ (visited on 11/04/2025).

C S, Apoorva et al. (July 2025). "Artificial Intelligence (AI) Driven Threat Detection and Mitigation using Machine Learning Techniques". In: *2025 8th International Conference on Computing Methodologies and Communication (ICCMC).* 2025 8th International Conference on Computing Methodologies and Communication (ICCMC), pp. 1629–1635. DOI: 10.1109/ICCMC65190.2025.11140747. URL: https://ieeexplore.ieee.org/document/11140747/ (visited on 11/04/2025).

Choudhary, Chahil et al. (Aug. 2025). "AI-Driven Threat Simulation in Wireless Networks using Advanced Machine Learning and Hyperparameter Optimization". In: *2025 5th International Conference on Soft Computing for Security Applications (ICSCSA).* 2025 5th International Conference on Soft Computing for Security Applications (ICSCSA), pp. 197–202. DOI: 10.1109/ICSCSA66339.2025.11170786. URL: https://ieeexplore.ieee.org/document/11170786/ (visited on 11/04/2025).

Edouard Bouda, R. Guy, Abdoulaye Sere, and Frederic Ouedraogo (Aug. 2025). "Improved Domain Name System Threat Detection using an RPZ and DGA-based approach incorporating Machine Learning". In: *2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA).* 2025 International Conference on Artificial Intelligence, Computer, Data Sciences and Applications (ACDSA), pp. 1–6. DOI: 10.1109/ACDSA65407.2025.11166262. URL: https://ieeexplore.ieee.org/document/11166262/ (visited on 11/04/2025).

El-Hajj, Mohammed (May 2025). "AI-Powered Threat Detection and Response: Leveraging Machine Learning for Real-Time Intrusion Detection Systems (IDS) Using Network Traffic Data". In: *2025 5th Intelligent Cybersecurity Conference (ICSC).* 2025 5th Intelligent Cybersecurity Conference (ICSC), pp. 84–90. DOI: 10.1109/ICSC65596.2025.11139927. URL: https://ieeexplore.ieee.org/document/11139927/ (visited on 11/04/2025).

Manikandan, K P, Navaneeswar Reddy Onteddu, and Abdul Kalam Chilimi (Mar. 2025). "Next-Gen Malware Detection using AI AI-Powered Malware Threat Detection Automated Malware Classification Through Machine Learning". In: *2025 International Conference on Intelligent Computing and Control Systems (ICICCS)*. 2025 International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 581–588. DOI: 10.1109/ICICCS65191.2025.10985389. URL: https://ieeexplore.ieee.org/document/10985389/ (visited on 11/04/2025).

Pandey, Pawan et al. (Aug. 2025). "AI-Powered Defenses: A Machine Learning Approaches in Cybersecurity Threat Detection". In: *2025 8th International Conference on Circuit, Power & Computing Technologies (ICCPCT).* 2025 8th International Conference on Circuit, Power & Computing Technologies (ICCPCT), pp. 394–399. DOI: 10.1109/ICCPCT65132.2025.11176700. URL: https://ieeexplore.ieee.org/document/11176700/ (visited on 11/04/2025).

Polinati, Anjani Kumar (Aug. 2025). "AI and Deep Learning-Powered Threat Intelligence and Automated Response Mechanisms". In: *2025 3rd International Conference on Sustainable Computing and Data Communication Systems (ICSCDS).* 2025 3rd International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), pp. 1504–1509. DOI: 10.1109/ICSCDS65426.2025.11167069. URL: https://ieeexplore.ieee.org/document/11167069/ (visited on 11/04/2025).

S, Rakesh et al. (Apr. 2025). "Neural Cybersecurity AI-Driven Deep Learning for Real-Time Threat Detection and Autonomous Defence". In: *2025 International Conference on Metaverse and Current Trends in Computing (ICMCTC).* 2025 International Conference on Metaverse and Current Trends in Computing (ICMCTC), pp. 1–4. DOI: 10.1109/ICMCTC62214.2025.11196383. URL: https://ieeexplore.ieee.org/document/11196383/ (visited on 11/04/2025).

Srilakshmi, P et al. (Apr. 2025). "Real-Time IoT Cybersecurity using Machine Learning-based AI Threat Detection System to Train Generative Robots". In: *2025 5th International Conference on Trends in Material Science and Inventive Materials (ICTMIM).* 2025 5th International Conference on Trends in Material Science and Inventive Materials (ICTMIM), pp. 1124–1130. DOI: 10.1109/ICTMIM65579.2025.10988029. URL: https://ieeexplore.ieee.org/document/10988029/ (visited on 11/04/2025).